

Multi-modality in augmented Lagrangian coordination for distributed optimal design

S. Tosserams · L. F. P. Etman · J. E. Rooda

Received: 26 September 2007 / Revised: 29 January 2009 / Accepted: 6 February 2009 / Published online: 26 March 2009
© The Author(s) 2009. This article is published with open access at Springerlink.com

Abstract This paper presents an empirical study of the convergence characteristics of augmented Lagrangian coordination (ALC) for solving multi-modal optimization problems in a distributed fashion. A number of test problems that do not satisfy all assumptions of the convergence proof for ALC are selected to demonstrate the convergence characteristics of ALC algorithms. When only a local search is employed at the subproblems, local solutions to the original problem are often attained. When a global search is performed at subproblems, global solutions to the original, non-decomposed problem are found for many of the examples. Although these findings are promising, ALC with a global subproblem search may yield only local solutions in the case of non-convex coupling functions or disconnected feasible domains. Results indicate that for these examples both the starting point and the sequence in which subproblems are solved determines which solution is obtained. We illustrate that the main cause for this behavior lies in the alternating minimization inner loop, which is inherently of a local nature.

Keywords Multidisciplinary design optimization · Decomposition · Augmented Lagrangian coordination · Non-convex · Multi-modal · Global optimization · Test problems

1 Introduction

The field of multidisciplinary design optimization (MDO) is concerned with the design of large-scale engineering systems that consist of a number of interacting subsystems. The size and the required level of expertise of each subsystem often prohibits the design of these large-scale systems to be performed in an integrated fashion. Instead, the problem is decomposed into smaller, more manageable parts, which are referred to as design subproblems. To deal with the resulting coupled subproblems, a systematical coordination approach to system design is required.

Many coordination methods have been proposed for distributed system optimization, and reviews are given in Cramer et al. (1994), Balling and Sobieszczanski-Sobieski (1996), and Tosserams et al. (2009b). These coordination methods include concurrent subspace optimization (CSSO, Sobieszczanski-Sobieski 1988), bi-level integrated system synthesis (BLISS/BLISS2000, Sobieszczanski-Sobieski et al. 2000, 2003), collaborative optimization (CO, Braun 1996; Braun et al. 1997), and the quasiseparable decomposition (QSD) approach of Haftka and Watson (2005). Several of these coordination methods may experience numerical difficulties when solving the master problem due to non-smoothness or failure to meet certain constraint qualifications (Alexandrov and Lewis 2002; DeMiguel and Murray 2000; Lin 2004). This may hinder the use of existing efficient gradient-based solution algorithms such as sequential quadratic programming (SQP). Methods that do not satisfy these requirements have to use specialized, typically inefficient algorithms to solve the associated optimization problems. Sobieszczanski-Sobieski et al. (2003); Sobieski and Kroo (2000); Liu

This work is funded by MicroNed, grant number 10005898.

S. Tosserams (✉) · L. F. P. Etman · J. E. Rooda
Department of Mechanical Engineering,
Eindhoven University of Technology,
PO Box 513, 5600 MB Eindhoven, The Netherlands
e-mail: s.tosserams@tue.nl

et al. (2004) propose the use of response surfaces to circumvent the difficulties due to the nonsmoothness.

During the last years, several new penalty function-based coordination methods have been developed: analytical target cascading (ATC) (Michelena et al. 1999; Kim 2001; Kim et al. 2003; Michelena et al. 2003; Tosserams et al. 2006), the inexact and exact penalty decomposition methods (IPD/EPD) of DeMiguel and Murray (2006), and the augmented Lagrangian coordination method for quasiseparable problems¹ (ALC-QS) of Tosserams et al. (2007). For these methods, basic constraint qualifications hold, and the optimization (sub)problems are smooth. All methods can be shown to converge to the optimal solution of the original problem under certain assumptions such as smoothness and convexity. The formulation of IPD/EPD is nested, similar to CO, and for every function evaluation of the master problem, an optimization of the subproblems is necessary. ATC and ALC-QS follow an alternating approach that iterates between solving the master problem and the disciplinary subproblems in a Gauss-Seidel fashion.

Recently, we demonstrated that a combination of augmented Lagrangian and Gauss-Seidel techniques can be used for the coordination of problems with both linking variables and coupling functions (Tosserams et al. 2008). This augmented Lagrangian coordination approach (ALC) has been demonstrated to provide flexibility to a system designer to tailor the formulation to existing organizational or computational structure. Analytical target cascading and ALC-QS are subclasses of the method.

Solutions obtained with ALC can be demonstrated to converge to Karush-Kuhn-Tucker (KKT) points of the original, non-decomposed problem under smoothness and convexity assumptions by combining existing results from the nonlinear programming community. Especially the assumption of convexity is difficult to enforce for practical design problems, which are typically non-convex and multi-modal (i.e. have multiple local solutions). Even when subproblems could be solved for global optimality (which for non-convex problems is very difficult, or even impossible to guarantee), then the ALC iterates may still not converge to a local solution (a KKT point) of the non-decomposed problem.

The objective of this study is to investigate how augmented Lagrangian coordination algorithms perform for a number of non-convex, multi-modal test problems

(i.e. problems with multiple local solutions) that do not satisfy all assumptions for convergence. We investigate whether the quality of solutions of ALC can be improved by using a global search at subproblems instead of a local search. Since a global search is in general far more expensive than a local search, the computational cost for both search options are compared as well. Moreover, the solutions obtained for both cases (local and global) are compared to an all-in-one implementation in terms of solution quality and computational cost required to obtain these solutions. This comparison is used to investigate whether the often-made suggestion that decomposition methods may be successful in reducing the computational cost when compared to integrated approaches holds for ALC.

The paper is outlined as follows. We first present the augmented Lagrangian method in Section 2, followed by a description of the general set-up of the numerical experiments in Section 3. Six examples and the associated numerical results are presented in Section 4, followed by a discussion of the general observations in Section 5. Finally, some concluding remarks are offered in Section 6.

2 Augmented Lagrangian coordination

The augmented Lagrangian coordination (ALC) method as presented in Tosserams et al. (2008) is concerned with solving the following system design problem:

$$\begin{aligned} \min_{\mathbf{z}} \quad & f_0(\mathbf{y}, \mathbf{x}_1, \dots, \mathbf{x}_M) + \sum_{j=1}^M f_j(\mathbf{y}, \mathbf{x}_j) \\ \text{subject to} \quad & \mathbf{g}_0(\mathbf{y}, \mathbf{x}_1, \dots, \mathbf{x}_M) \leq \mathbf{0} \\ & \mathbf{h}_0(\mathbf{y}, \mathbf{x}_1, \dots, \mathbf{x}_M) = \mathbf{0} \\ & \mathbf{g}_j(\mathbf{y}, \mathbf{x}_j) \leq \mathbf{0} \quad j = 1, \dots, M \\ & \mathbf{h}_j(\mathbf{y}, \mathbf{x}_j) = \mathbf{0} \quad j = 1, \dots, M \end{aligned} \quad (1)$$

where $\mathbf{z} = [\mathbf{y}, \mathbf{x}_1, \dots, \mathbf{x}_M]$

where M is the number of subsystems. The vector of design variables $\mathbf{z} = [\mathbf{y}, \mathbf{x}_1, \dots, \mathbf{x}_M]$ consists of a number of linking variables \mathbf{y} , and a number of local variables \mathbf{x}_j associated exclusively to subsystem j . The linking variables may be common design variables shared by multiple subsystems, and interdisciplinary coupling variables that link the analysis models of different subsystems. The coupling objective f_0 and coupling constraints \mathbf{g}_0 and \mathbf{h}_0 are non-separable and may depend on all design variables \mathbf{z} . Local objectives f_j , and local constraints \mathbf{g}_j and \mathbf{h}_j are associated exclusively to subsystem j , and may depend on the linking variables \mathbf{y} and the local variables \mathbf{x}_j of only a single subsystem j .

¹Quasiseparable problems have subsystems that are only coupled through linking variables; coupling objectives and constraints are not allowed.

In this paper we are particularly interested in multi-modal problems, i.e. problems that have multiple local solutions. Let $f = f_0 + f_1 + \dots + f_M$, $\mathbf{g} = [\mathbf{g}_0, \mathbf{g}_1, \dots, \mathbf{g}_M]$, and $\mathbf{h} = [\mathbf{h}_0, \mathbf{h}_1, \dots, \mathbf{h}_M]$, then the feasible domain \mathcal{F} of (1) is defined by $\mathcal{F} = \{\mathbf{z} | \mathbf{g}(\mathbf{z}) \leq \mathbf{0}, \mathbf{h}(\mathbf{z}) = \mathbf{0}\}$. Furthermore, let $\mathcal{B}(\mathbf{z}, \varepsilon)$ be the ball with radius $\varepsilon > 0$ around \mathbf{z} . Then, a feasible point $\mathbf{z}^* \in \mathcal{F}$ is a local solution of problem (1) if it satisfies

$$f(\mathbf{z}^*) \leq f(\mathbf{z}) \quad \forall \mathbf{z} \in \mathcal{F} \cap \mathcal{B}(\mathbf{z}^*, \varepsilon) \quad (2)$$

for some positive value of ε . In words, a local solution \mathbf{z}^* is a feasible point with the lowest objective when compared to the other feasible points in its direct neighborhood. If multiple of these local solutions exist, then a problem is multi-modal. The global optimum $\mathbf{z}_g^* \in \mathcal{F}$ is defined as the point with the lowest objective in the entire feasible domain, and therefore satisfies

$$f(\mathbf{z}_g^*) \leq f(\mathbf{z}) \quad \forall \mathbf{z} \in \mathcal{F} \quad (3)$$

For continuous problems, multi-modality often arises due to the non-convexity of objective and/or constraint functions.

2.1 Decomposition of the original problem

ALC introduces three steps to decompose problem (1). In the first step, coupling of the constraints through the linking variables \mathbf{y} is removed. A separate copy \mathbf{y}_j is introduced at each subsystem, such that the local constraints \mathbf{g}_j and \mathbf{h}_j depend on the copy \mathbf{y}_j associated with subsystem j and the local variables \mathbf{x}_j of this subsystem. The copies are forced equal (i.e. $\mathbf{y}_1 = \mathbf{y}_2 = \dots = \mathbf{y}_M$) by consistency constraints \mathbf{c} , which are defined as a collection of consistency constraints \mathbf{c}_{jn} between subsystem j and its neighbors $n \in \mathcal{N}_j$:

$$\mathbf{c}_{jn} = \mathbf{y}_j - \mathbf{y}_n = \mathbf{0} \quad \{n \in \mathcal{N}_j | n > j\} \quad j = 1, \dots, M \quad (4)$$

The neighbors \mathcal{N}_j are defined as the subsystems to which subsystem j is linked to through the consistency constraints. Furthermore, the condition $n > j$ makes sure that only one of the linearly dependent pair \mathbf{c}_{jn} and \mathbf{c}_{nj} is included in the consistency constraints (e.g., only $\mathbf{c}_{12} = \mathbf{y}_1 - \mathbf{y}_2$, and not also $\mathbf{c}_{21} = \mathbf{y}_2 - \mathbf{y}_1$). For a detailed discussion on consistency constraint allocation, the reader is referred to Allison and Papalambros (2008).

The second step involves relaxation of the consistency constraints \mathbf{c} as well as the coupling constraints \mathbf{g}_0 and \mathbf{h}_0 . The vector $\mathbf{q} = [\mathbf{c}, \mathbf{g}_0 + \mathbf{s}, \mathbf{h}_0]$ is introduced as the vector of all linking constraints, where $\mathbf{s} \geq \mathbf{0}$ are

slack variables that allow the inequality constraints \mathbf{g}_0 to be treated as equality constraints. All constraints \mathbf{q} are relaxed using an augmented Lagrangian penalty function (see, e.g., Bertsekas 1982; Arora et al. 1991; Bertsekas 2003)

$$\phi(\mathbf{q}) = \mathbf{v}^T \mathbf{q} + \|\mathbf{w} \circ \mathbf{q}\|_2^2 \quad (5)$$

Here, \mathbf{v} and \mathbf{w} are penalty parameters that need to be selected appropriately. The symbol \circ represents the Hadamard product: an entry-wise multiplication of two vectors, such that $\mathbf{a} \circ \mathbf{b} = [a_1, \dots, a_n]^T \circ [b_1, \dots, b_n]^T = [a_1 b_1, \dots, a_n b_n]^T$. After relaxation of the linking constraints, subsystems have separable constraint sets, and are only linked through the objective function.

The third step of decomposition is to define a subproblem P_j for each subsystem. The augmented Lagrangian coordination algorithms alternately optimize for the variables of each subsystem. Subproblem P_j therefore only includes the terms that depend on its variables $\bar{\mathbf{x}}_j = [\mathbf{x}_j, \mathbf{y}_j, \mathbf{s}_j]$, and is given by

$$\begin{aligned} \min_{\bar{\mathbf{x}}_j} & f_j(\mathbf{y}_j, \mathbf{x}_j) + f_0(\mathbf{y}_1, \mathbf{x}_1, \dots, \mathbf{y}_M, \mathbf{x}_M) \\ & + \phi(\mathbf{q}(\mathbf{y}_1, \mathbf{x}_1, \dots, \mathbf{y}_M, \mathbf{x}_M, \mathbf{s})) \\ \text{subject to} & \mathbf{g}_j(\mathbf{y}_j, \mathbf{x}_j) \leq \mathbf{0} \\ & \mathbf{h}_j(\mathbf{y}_j, \mathbf{x}_j) = \mathbf{0} \\ & \mathbf{s}_j \geq \mathbf{0} \\ \text{where } \bar{\mathbf{x}}_j &= [\mathbf{y}_j, \mathbf{x}_j, \mathbf{s}_j] \end{aligned} \quad (6)$$

where \mathbf{s}_j is the (possibly empty) vector of slack variables allocated to subsystem j such that $[\mathbf{s}_1, \dots, \mathbf{s}_M]$ holds the same elements as \mathbf{s} but not necessarily in the same order.

The solution to subproblem P_j depends on the solution of the other subproblems $P_i | i \neq j$, since the variables of these subsystems appear in the objective of P_j . A coordination algorithm is necessary to account for this coupling.

2.2 Coordination algorithms

Coordination algorithms for ALC have two tasks:

1. To select appropriate penalty parameters \mathbf{v} and \mathbf{w}
2. To account for the coupling of subproblem objectives

Two common-place nonlinear programming techniques are combined to perform these tasks in a coordination algorithm that consists of inner and outer loops. The method of multipliers (see, e.g. Bertsekas 2003) is used in the outer loop to set the penalty parameters, and an alternating minimization approach (see, e.g. Bezdek

and Hathaway 2002) that sequentially solves the subproblems (6) $j = 1, \dots, M$ accounts for the subproblem coupling in an inner loop. The coordination algorithm is illustrated in Fig. 1a.

2.2.1 Outer loop: method of multipliers

In the outer loops, the method of multipliers sets the penalty parameters \mathbf{v}^{k+1} for outer iteration $k + 1$ using the following update formula (see, e.g., Bertsekas 1982, 2003):

$$\mathbf{v}^{k+1} = \mathbf{v}^k + 2\mathbf{w}^k \circ \mathbf{w}^k \circ \mathbf{q}^k \quad (7)$$

where \mathbf{q}^k are the values of the linking constraints \mathbf{q} at termination of the k -th inner loop. Since large penalty weights slow down the coordination algorithms and introduce ill-conditioning of the subproblems, the penalty weights \mathbf{w} are increased a factor β only when the reduction in the linking constraint value is smaller than some fraction γ . If the reduction is larger, the penalty weights are not updated. As a result, the penalty weights are only increased when the contribution of the Lagrange multiplier update (7) did not lead to a large enough reduction in the violation of the linking constraints. More formally, the penalty weight w_i for the i th linking constraint q_i is updated as (Bertsekas 2003)

$$w_i^{k+1} = \begin{cases} w_i^k & \text{if } |q_i^k| \leq \gamma |q_i^{k-1}| \\ \beta w_i^k & \text{if } |q_i^k| > \gamma |q_i^{k-1}| \end{cases} \quad (8)$$

where $\beta > 1$ and $0 < \gamma < 1$, and we observe that $\beta = 2.2$ and $\gamma = 0.4$ perform well in general.

The outer loop, and thus the solution procedure, is terminated when two conditions are satisfied. First, the

change in the maximal linking constraint value for two consecutive outer loop iterations must be smaller than some user-defined termination tolerance $\varepsilon > 0$

$$\|\mathbf{q}^k - \mathbf{q}^{k-1}\|_\infty < \varepsilon \quad (9)$$

Second, the maximal linking constraint violation must also be smaller than tolerance $\varepsilon > 0$

$$\|\mathbf{q}^k\|_\infty < \varepsilon \quad (10)$$

Note that it is also possible to use different termination tolerances for each criterion, however for the sake of simplicity we use a single value for both criteria.

2.2.2 Inner loop: alternating optimization

In the inner loop, subproblems are solved sequentially for fixed penalty parameters \mathbf{v} and \mathbf{w} using an alternating optimization approach (see, e.g., Fortin and Glowinski 1983; Grippo and Sciandrone 2000; Bezdek and Hathaway 2002).² This procedure is terminated when the relative change in the objective function value F of the relaxed system design problem given by

$$F(\mathbf{x}_1, \mathbf{y}_1, \dots, \mathbf{x}_M, \mathbf{y}_M, \mathbf{s}) = \sum_{j=1}^M f_j(\mathbf{x}_j, \mathbf{y}_j) + f_0(\mathbf{x}_1, \mathbf{y}_1, \dots, \mathbf{x}_M, \mathbf{y}_M) + \phi(\mathbf{q}(\mathbf{x}_1, \mathbf{y}_1, \dots, \mathbf{x}_M, \mathbf{y}_M, \mathbf{s}))$$

for two consecutive inner loop iterations is smaller than some user-defined termination tolerance $\varepsilon_{\text{inner}} > 0$:

$$\frac{|F^\xi - F^{\xi-1}|}{1 + |F^\xi|} < \varepsilon_{\text{inner}} \quad (11)$$

where ξ denotes the inner loop iteration number. The division by $1 + |F^\xi|$ is used for proper scaling of the criterion for very large as well as very small objectives (Gill et al. 1981). The termination tolerance $\varepsilon_{\text{inner}}$ should be smaller than the outer loop termination tolerance ε to assure sufficient accuracy of the inner loop solution. We use $\varepsilon_{\text{inner}} = \varepsilon/100$.

An alternative inner loop termination strategy is to cut off the inner loop before actual convergence during

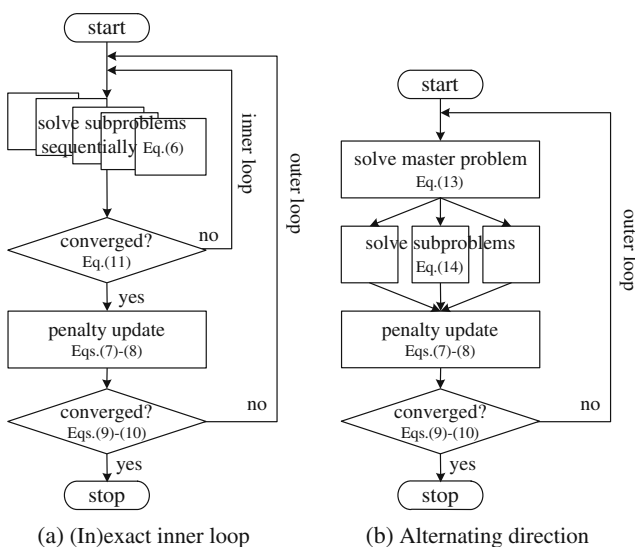


Fig. 1 Illustration of coordination algorithms for ALC (a, b)

²The alternating optimization approach is also known as “non-linear Gauss-Seidel” or “block-coordinate descent” (see, e.g., Bertsekas and Tsitsiklis 1989; Bertsekas 2003).

the first few iterations by using looser tolerances. More formally, such an inexact approach uses a different tolerance $\varepsilon_{\text{inner}}^k$ for each outer loop iteration. The main idea behind such a strategy is that costly inner loop iterations are avoided when the penalty parameters are still far from their optimal values. Convergence for the outer loop updates in case of an inexact inner loop can still be guaranteed, as long as the sequence of termination tolerances $\{\varepsilon_{\text{inner}}^k\}$ is non-increasing, and $\varepsilon_{\text{inner}}^k \rightarrow 0$ (Bertsekas 2003).

2.3 Subclasses using an alternating direction approach

The alternating direction method of multipliers of Bertsekas and Tsitsiklis (1989) has been proposed to coordinate several subclasses of decomposed ALC problems (see, e.g., Tosserams et al. 2006, 2007, 2009a; Li et al. 2008). Coordination algorithms based on an alternating direction approach require only a single iteration in the inner loop. The most general of these classes considers block-dependent coupling functions. A function f is block-dependent if it depends on disciplinary response functions $\mathbf{r}_j(\mathbf{y}, \mathbf{x}_j)$ of different subsystems $j = 1, \dots, M$, but the functions \mathbf{r}_j themselves are not separable in \mathbf{y} and \mathbf{x}_j such that $f(\mathbf{r}_1(\mathbf{y}, \mathbf{x}_1), \dots, \mathbf{r}_M(\mathbf{y}, \mathbf{x}_M))$. These problems are defined as

$$\begin{aligned} \min_{\mathbf{z}} \quad & f_0(\mathbf{r}_1(\mathbf{y}, \mathbf{x}_1), \dots, \mathbf{r}_M(\mathbf{y}, \mathbf{x}_M)) + \sum_{j=1}^M f_j(\mathbf{y}, \mathbf{x}_j) \\ \text{subject to} \quad & \mathbf{g}_0(\mathbf{r}_1(\mathbf{y}, \mathbf{x}_1), \dots, \mathbf{r}_M(\mathbf{y}, \mathbf{x}_M)) \leq \mathbf{0} \\ & \mathbf{h}_0(\mathbf{r}_1(\mathbf{y}, \mathbf{x}_1), \dots, \mathbf{r}_M(\mathbf{y}, \mathbf{x}_M)) = \mathbf{0} \\ & \mathbf{g}_j(\mathbf{y}, \mathbf{x}_j) \leq \mathbf{0} \quad j = 1, \dots, M \\ & \mathbf{h}_j(\mathbf{y}, \mathbf{x}_j) = \mathbf{0} \quad j = 1, \dots, M \\ \text{where } \mathbf{z} = & [\mathbf{y}, \mathbf{x}_1, \dots, \mathbf{x}_M] \end{aligned} \quad (12)$$

where \mathbf{r}_j is the vector of component functions of subsystem j . The approach for block-dependent problems introduces local copies \mathbf{y}_j , $j = 1, \dots, M$ of the linking variables \mathbf{y} , but keeps the original \mathbf{y} as a “master copy”. The consistency constraints are then defined as $\mathbf{c}_{0j} = \mathbf{y} - \mathbf{y}_j = \mathbf{0}$, $j = 1, \dots, M$. Similarly, support variables \mathbf{t}_j are introduced for each vector of component functions \mathbf{r}_j , as well as consistency constraints $\mathbf{c}_j^t = \mathbf{t}_j - \mathbf{r}_j(\mathbf{y}_j, \mathbf{x}_j) = \mathbf{0}$ that force the support variables to be equal to the original component functions. These support variables assume the role of the component functions in the coupling functions such that $f_0 = f_0(\mathbf{t}_1, \dots, \mathbf{t}_M)$, $\mathbf{g}_0 = \mathbf{g}_0(\mathbf{t}_1, \dots, \mathbf{t}_M)$, and $\mathbf{h}_0 = \mathbf{h}_0(\mathbf{t}_1, \dots, \mathbf{t}_M)$. The set of linking constraints becomes $\mathbf{q} = [\mathbf{c}, \mathbf{c}^t]$, which is relaxed using an augmented

Lagrangian penalty ϕ . The relaxed problem is then decomposed into $M + 1$ subproblems. A master problem P_0 optimizes for \mathbf{y} and $\mathbf{t}_1, \dots, \mathbf{t}_M$, and the M disciplinary problems P_j solve for \mathbf{y}_j and \mathbf{x}_j . The master problem P_0 is given by

$$\begin{aligned} \min_{\mathbf{y}, \mathbf{t}_1, \dots, \mathbf{t}_M} \quad & f_0(\mathbf{t}_1, \dots, \mathbf{t}_M) \\ & + \sum_{j=1}^M [\phi(\mathbf{y} - \mathbf{y}_j) + \phi(\mathbf{t}_j - \mathbf{r}_j(\mathbf{y}_j, \mathbf{x}_j))] \\ \text{subject to} \quad & \mathbf{g}_0(\mathbf{t}_1, \dots, \mathbf{t}_M) \leq \mathbf{0} \\ & \mathbf{h}_0(\mathbf{t}_1, \dots, \mathbf{t}_M) = \mathbf{0} \end{aligned} \quad (13)$$

The disciplinary subproblems P_j , $j = 1, \dots, M$ are given by

$$\begin{aligned} \min_{\mathbf{y}_j, \mathbf{x}_j} \quad & f_j(\mathbf{y}_j, \mathbf{x}_j) + \phi(\mathbf{y} - \mathbf{y}_j) + \phi(\mathbf{t}_j - \mathbf{r}_j(\mathbf{y}_j, \mathbf{x}_j)) \\ \text{subject to} \quad & \mathbf{g}_j(\mathbf{y}_j, \mathbf{x}_j) \leq \mathbf{0} \\ & \mathbf{h}_j(\mathbf{y}_j, \mathbf{x}_j) = \mathbf{0} \end{aligned} \quad (14)$$

An attractive property of the above formulation is that the objectives of the disciplinary subproblems are only coupled to the objective of the master problem, and not to each other. This allows the inner loop to consist of two phases: a first phase in which the master problem P_0 is solved, and a second phase in which the disciplinary subproblems P_j , $j = 1, \dots, M$ are solved in parallel (see Fig. 1b).

Note that the ALC variant for block-separable linking constraints of Tosserams et al. (2009a) is a subclass of the above formulation, as well as the ALC variant for quasiseparable problems of Tosserams et al. (2007). In the first case, the coupling functions depend linearly on the component functions, and the master problem becomes a quadratic programming problem. In the latter case, no coupling functions are present at all, and the master problem reduces to the minimization of a convex quadratic function, to which an analytical solution is available.

2.4 Convergence properties

The solutions obtained with the ALC algorithms that have an iterative inner loop have been demonstrated to be KKT points of the original non-decomposed problem (1) when the objective and coupling constraints are smooth, and when subproblem constraints are convex (Tosserams et al. 2008). These convergence results are derived from existing convergence theory for the method of multipliers as given in Bertsekas (2003) and for the alternating minimization technique (Grippio and Sciandrone 2000; Bezdek and Hathaway 2002; Bertsekas 2003). Whether the obtained stationary

points are close to the global optimum is however not known since the penalty update and alternating optimization methods are local search methods. This article therefore aims at investigating the convergence behavior of ALC for non-convex problems with multiple local solutions.

The convergence proofs associated with the alternating direction method of multipliers variants as given in Bertsekas and Tsitsiklis (1989) assumes that both the objective and constraints of problem (1) are smooth and convex. Again, we are interested in the performance of these special alternating direction variants of ALC on the selected non-convex and multi-modal examples. More moderate values of β and γ are advised for the alternating direction approaches, and $\beta = 1.2$ and $\gamma = 0.75$ appear to work well in general. Note that the alternating direction formulation can also be implemented with an iterative inner loop, similar to the ALC algorithms presented earlier in this section.

3 Numerical experiments setup

We investigate six test problems in this paper. Similar experiments are set up for each of the test problems of Section 4. First, the problems are solved in an all-in-one implementation using the LGO global search solver implemented in TOMLAB (LGO Solver Suite 2008) under default settings and a termination tolerance of 10^{-2} . In its default settings, the LGO solver first performs an adaptive random global search through the design space to identify a number of promising points from each of which a local search is initiated. This global search is repeated ten times for each problem to obtain an estimate of the number of function evaluations required. Alternatively, stochastic global search algorithms such as genetic algorithms, particle swarm, or simulation annealing may be used. We selected the LGO solver in this study because of the quality and reproducibility of results, although it may be less efficient than other global search algorithms. A detailed performance comparison between different global and local search methods in the context of decomposition, although relevant, is beyond the scope of this paper.

The computational cost of the all-in-one solution serves as a reference to which the computational costs for the decomposed experiments can be compared. To get an impression of the distribution of the local solutions, each test problem is solved in an all-in-one formulation with MatLab's SQP solver `fmincon` (Mathworks 2008) under default settings from 1000 different starting points selected randomly within the bounds.

Second, one or more partitions of each test problem is selected, and coordinated with an ALC implementation. For all partitions, even those with coupling functions, an alternating direction ALC approach is used that uses only a single iteration inner loop. Although the convergence proof is not valid for non-convex problems, the implementation is very relevant from a practical perspective since the alternating direction method has been found to be more efficient for many examples than a nested inner loop implementation (Tosserams et al. 2006, 2007, 2009a; Li et al. 2008).

The penalty weights are updated using (8) with $\beta = 1.2$ and $\gamma = 0.75$. Similar to Tosserams et al. (2008), we initially set $\mathbf{v} = \mathbf{0}$, and take all weights equal $\mathbf{w} = w$, such that $\phi = w^2 \mathbf{q}^T \mathbf{q}$ where $\mathbf{q} = [\mathbf{c}, \mathbf{g}_0 + \mathbf{s}, \mathbf{h}_0]$ are the linking constraints. The initial weights are selected as

$$w = \sqrt{\frac{\alpha |\hat{f}|}{\hat{\mathbf{q}}^T \hat{\mathbf{q}}}} \quad (15)$$

to make the penalty terms and the objective of comparable magnitude. Here, \hat{f} and $\hat{\mathbf{q}}$ are estimates of typical objective function and linking constraint values. For each example, the typical objective is $\hat{f} = 1$, except for the second example that uses $\hat{f} = -100$. The estimates for the linking constraints $\hat{\mathbf{q}}$ are obtained by solving the decomposed problems for small weights $\mathbf{w} = 10^{-3}$, and zero Lagrange multipliers $\mathbf{v} = \mathbf{0}$. For these weights, the penalty terms will be small when compared to the objective function value. As a consequence, the allowed linking constraint violations will be large, and the solution of the relaxed problem will produce an estimate $\hat{\mathbf{q}}_j$ for the size of the linking constraint values.

Two subproblem solution strategies are used to investigate their influence on the performance of the ALC algorithms. First, subproblems are solved using the MatLab's local search SQP algorithm `fmincon` under default settings. Second, a global search is performed at subproblems using the multi-start LGO global search. The first variant is started from 100 different starting points selected randomly within the bounds, and for the second 10 random starting points are selected. For each of these options, the obtained solutions, average number of required function evaluations, and the average number of required subproblem optimizations are reported as comparative metrics. The computational cost measures are taken as the average over all converged starting points.

The termination tolerance for the outer loop is set to $\varepsilon = 10^{-2}$, while the termination tolerance for the subproblems is set to $\varepsilon/10^4 = 10^{-6}$ to assure sufficient accuracy.

4 Results

In this section, numerical results for six examples are presented. A discussion of the general observations based on these results is offered in Section 5.

4.1 Example 1

The first example demonstrates that the solutions obtained with ALC correspond to the solutions found with an all-in-one approach. The example has fourteen variables, four inequality constraints, and three equality constraints. The objective is convex, but the constraints are non-convex, therefore violating the assumptions of the convergence proof of the augmented Lagrangian coordination algorithms. The problem is given by

$$\begin{aligned} \min_{z_1, \dots, z_{14}} f &= \sum_{i=1}^{14} F_i(z_i) = \sum_{i=1}^{14} (z_i - 3)^2 \\ \text{s.t. } g_1 &= (\sin(z_3^{-2}) + z_4^2)z_5^{-2} - 1 \leq 0 \\ g_2 &= (z_8^2 + z_9^2)z_{11}^{-2} - 1 \leq 0 \\ g_3 &= (z_{11}^2 + \sin(z_{12}^{-2}))z_{13}^{-2} - 1 \leq 0 \\ g_4 &= (z_5^2 + z_6^{-2})z_7^{-2} + (z_8^{-2} + z_{10}^2)z_{11}^{-2} \\ &\quad + (z_{11}^2 + z_{12}^2)z_{14}^{-2} - 3 \leq 0 \\ h_1 &= (z_3^2 + z_4^{-2} + z_5^2)z_1^2 - 1 = 0 \\ h_2 &= (\sin(z_{11}^2) + z_{12}^2 + z_{13}^2 + z_{14}^2)\sin(z_6^2) - 1 = 0 \\ h_3 &= (z_5^2 + z_6^2 + z_7^2)z_2^2 + (z_8^2 + z_9^{-2} + z_{10}^{-2} + z_{11}^2)z_3^{-2} \\ &\quad - 2 = 0 \\ 0.1 &\leq z_i \leq 5, \quad i = 1, \dots, 14 \end{aligned} \quad (16)$$

All 10 runs of the all-in-one experiments using the global search variant of LGO converged to an optimal value (rounded) of $f^* = 17.56$, with $\mathbf{z}^* = [0.23, 0.13, 3.26,$

$2.78, 2.80, 3.06, 3.21, 2.25, 2.38, 2.73, 3.27, 2.76, 3.29, 3.47]$ and all constraints active. The average number of function evaluations reported by LGO was 447,455. The 1000 all-in-one runs with the Matlab's local search SQP implementation `fmincon` converged to a large number of local minima, including the one obtained with the global search. Table 1 reports the local minima that were obtained from at least ten of the starting points. The solutions obtained less than ten times have optimal values scattered between 19.59 and 45.98, and are reported as "other" in Table 1. This scattering is most likely caused by numerical errors caused by finite differencing and termination tolerances. The `fmincon` solver did not converge to a solution from 50 of the 1000 starting points. The average number of function evaluations for the converged runs was 277.

Problem (16) is partitioned into four subsystems that are coupled through linking variables $\mathbf{y} = [z_3, z_5, z_6, z_{11}]$ and coupling constraints $\mathbf{g}_0 = [g_4]$ and $\mathbf{h}_0 = [h_3]$. Subsystem 1 has local variables $\mathbf{x}_1 = [z_1, z_4]$, local objective $f_1 = F_1 + F_3 + F_4 + F_5$, and local constraints $\mathbf{g}_1 = [g_1]$ and $\mathbf{h}_1 = [h_1]$. Subsystem 2 has local variables $\mathbf{x}_2 = [z_2, z_7]$, local objective $f_2 = F_2 + F_6 + F_7$, and no local constraints $\mathbf{g}_2 = []$ and $\mathbf{h}_2 = []$. Subsystem 3 has $\mathbf{x}_3 = [z_8, z_9, z_{10}]$, $f_3 = F_8 + F_9 + F_{10} + F_{11}$, and $\mathbf{g}_3 = [g_2]$ and $\mathbf{h}_3 = []$. Subsystem 4 has $\mathbf{x}_4 = [z_{12}, z_{13}, z_{14}]$, $f_4 = F_{12} + F_{13} + F_{14}$, and $\mathbf{g}_4 = [g_3]$ and $\mathbf{h}_4 = [h_2]$.

This partition is coordinated using two ALC variants. Both variants are depicted in Fig. 2 in which the boxes represent subproblems or master problems. For subproblems, a list of local variables \mathbf{x}_j , possible slack variables \mathbf{s}_j , and local functions is given within a box. For the master problem, the box lists the support variables \mathbf{t} and the system-wide functions that are included in the master problem. Solid lines between boxes represent consistency constraints, and are annotated by the corresponding linking variables \mathbf{y} , or by the consistency constraint \mathbf{c}_j^i in case of support variable-response function couplings that appear in (13)–(14).

Fig. 2 Example 1: two ALC coordination variants.

Solid lines indicate coupling through consistency constraints, and dashed lines indicate coupling through coupling constraints. The linking variables of the decomposed problem are $\mathbf{y} = [z_3, z_5, z_6, z_{11}]$ (a, b)

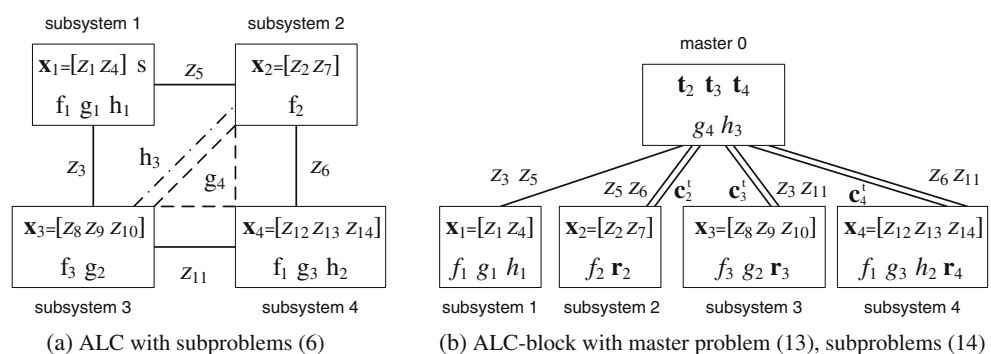


Table 1 Example 1

	Algorithm					
	fmincon	LGO	ALC		ALC block	
	Local	Global	Local	Global	Local	Global
$f^* = 17.56$	99	10	12	10	12	10
$f^* = 17.75$	129	0	19	0	20	0
$f^* = 17.89$	96	0	7	0	5	0
$f^* = 18.54$	84	0	15	0	15	0
$f^* = 19.00$	174	0	11	0	12	0
$f^* = 19.46$	87	0	3	0	2	0
$f^* = 20.52$	45	0	8	0	9	0
$f^* = 30.57$	221	0	25	0	25	0
Other $19.59 \leq f^* \leq 45.98$	15	0	0	0	0	0
NC	50	0	0	0	0	0
Total	1000	10	100	10	100	10
Subproblem optimizations	–	–	48	56	54	56
Function evaluations	277	447,455	1: 998 2: 1,700 3: 1,358 4: 1,114	1: 1,845,652 2: 267,907 3: 726,922 4: 1,796,240	1: 943 2: 1,964 3: 1,601 4: 1,288 0: 3,682	1: 1,383,293 2: 268,128 3: 732,471 4: 1,957,680 0: 19,851

Number of starting points that converged to a specific objective function value for different algorithms, average number of required function evaluations and subproblem optimizations. The function evaluations are listed per subproblem, where subproblem “0” is the convex master problem of the block-separable coordination variant. NC = did not converge to a feasible solution

Dashed lines represent coupling functions that link two or more subproblems.

The first variant, labeled ALC, coordinates the three subsystems in a traditional ALC fashion. The slack variable s for the coupling constraint \mathbf{g}_0 is assigned to Subsystem 1 such that $\mathbf{s}_1 = [s]$, and $\mathbf{s}_2 = \mathbf{s}_3 = \mathbf{s}_4 = []$.

The second variant uses the fact that the coupling constraints of the first partition become block-separable with $\mathbf{r}_2 = [(z_5^2 + z_6^{-2})z_7^{-2} - 1, (z_5^2 + z_6^2 + z_7^2)z_2^2 - 1]$, $\mathbf{r}_3 = [(z_8^{-2} + z_{10}^2)z_{11}^{-2} - 1, (z_8^2 + z_9^{-2} + z_{10}^{-2} + z_{11}^2)z_3^{-2} - 1]$ and $\mathbf{r}_4 = [(z_{11}^2 + z_{12}^2)z_{14}^{-2} - 1]$, such that $\mathbf{g}_0 = g_4 = r_{2,1} + r_{3,1} + r_{4,1}$ and $\mathbf{h}_0 = h_3 = r_{2,2} + r_{3,2}$. This second variant, ALC-block, includes a master problem that has support variables $\mathbf{t}_2 = [t_{2,1}, t_{2,2}]$, $\mathbf{t}_3 = [t_{3,1}, t_{3,2}]$, and $\mathbf{t}_4 = [t_{4,1}]$ as its local variables, and constraints g_4 and h_3 as local constraints.

Table 1 displays the results for the distributed optimization experiments. These results clearly show the ability of both ALC variants to converge to solutions of the all-in-one problem, even though some requirements for the convergence proof are not met. In general, the solutions obtained with a local search at subproblems are similar to those obtained with the all-in-one implementation using a local search. Strikingly, all experiments that performed a global search at the subproblems converged to the best obtained solution of the all-in-one implementation. The next examples however illustrate that this is not a general characteristic of ALC, but depends on the problem. The large

computational differences between the local and global search implementations illustrate that performing a global search is far more expensive than performing only a local search.

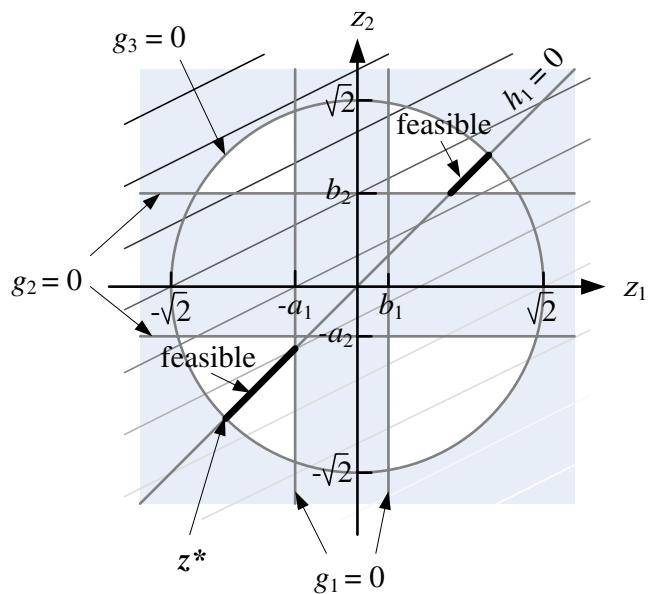


Fig. 3 Illustration of geometry of Example 2 for $m = 1$. Shaded areas are infeasible with respect to the inequality constraints, and the feasible domain is further constrained by h_1 to the line $z_1 = z_2$. Objective contours are dark for high values and light for low values. The optimal design is indicated by $\mathbf{z}^* = [-1, -1]$

The required number of function evaluations for ALC are for most subproblems a factor 2-10 higher than those observed for the all-in-one experiments, even though the solution cost for a single subproblem optimization is smaller than the cost for solving the all-in-one problem. Note that subproblem 2 with a global search is an exception since it shows a decrease in function evaluations when compared to the all-in-one results. The overhead introduced by coordination incurs additional cost, which makes the decomposition approach more costly than an all-in-one implementation. The differences in required function evaluations between the two ALC variants are relatively small, although the cost for the second partition could be reduced since its subproblems can be solved in parallel.

The large differences in computational cost between subproblems when using a global search are caused by the differences in the number of variables and constraints per subproblem. Subsystems 1 and 4 have five variables (three local and two shared) and two local constraints in the first ALC variant, while Subsystem 3 has only one constraint, and Subsystem 2 has only four variables and no constraints. A similar difference holds for the ALC-block variant. Note that the function evaluations for the convex master problem for ALC-block do not depend on whether subproblems are solved with a global search, since the master problem itself is always solved with the cheaper, local search variant of LGO.

4.2 Example 2

The second example problem shows that cost reductions through decomposition can be obtained for some problems, while preserving global optimality of solutions. The problem is given by

$$\begin{aligned} \min_{z_1, \dots, z_{2m}} f &= \sum_{i=1}^{2m} F_i = \sum_{i=1}^m 2z_i - \sum_{i=m+1}^{2m} z_i \\ \text{subject to } g_i &= -\left(z_i + a_i \sqrt{\frac{1}{m}}\right) \left(z_i - b_i \sqrt{\frac{1}{m}}\right) \leq 0 \\ &\quad i = 1, \dots, 2m \\ g_{2m+1} &= \sum_{i=1}^{2m} z_i^2 - 2 \leq 0 \\ h_i &= z_i - z_{i+m} = 0 \quad i = 1, \dots, m \\ -2 &\leq z_i \leq 2 \quad i = 1, \dots, 2m \end{aligned} \quad (17)$$

where m is a scaling parameter that determines the number of variables and constraints. $0 \leq a_i, b_i \leq 1$ are parameters that determine the feasible domain of the problem, but that do not affect the global solution. For

the numerical experiments, we take $a_i = b_i = \frac{1}{2}$. The global solution to this problem has all $z_i^* = -\sqrt{\frac{1}{m}}$, and $f^* = -\sqrt{m}$. The geometry of the feasible domain for $m = 1$ is illustrated in Fig. 3.

Problem (17) has 2^m local minima that can be determined analytically. First, $z_{i+m} = z_i$ for $i = 1, \dots, m$ follows directly from the equality constraints, reducing the dimension of the problem to m . By substituting these relations in the remaining functions, the objective simply becomes the sum of the remaining m variables. Constraints g_1, \dots, g_{2m} are non-convex and force variables z_i , $i = 1, \dots, m$ to be either smaller than $-\bar{a}_i = -\sqrt{1/m} \max(a_i, a_{i+m})$ or larger than $\bar{b}_i = \sqrt{1/m} \max(b_i, b_{i+m})$. The maximum terms appear because of the eliminated variables z_{m+1}, \dots, z_{2m} . Constraint g_{2m+1} limits the feasible domain further to an m -dimensional unit hypersphere. Since each of the m variables is either smaller than $-\bar{a}_i$ or greater than \bar{b}_i , the feasible domain consists of 2^m disconnected regions. Each of these regions has a single local minimizer yielding a total of 2^m local solutions to problem (17).

The 2^m local solutions can be divided into $m+1$ groups that have equal objective values. Group n has n variables larger than \bar{b}_i , and the remaining $m-n$ are smaller than $-\bar{a}_i$. Here $n = \{0, \dots, m\}$, and group $n = 0$ contains the global optimum. Let $\mathcal{J}_+ \subseteq \{1, \dots, m\}$ be the indices of the variables that are positive, where $|\mathcal{J}_+| = n$ is the number of elements in set \mathcal{J}_+ , and let $\mathcal{J}_- \subseteq \{1, \dots, m\}$ be the set of indices of the negative variables such that $\mathcal{J}_- \cup \mathcal{J}_+ = \{1, \dots, m\}$ and $\mathcal{J}_- \cap \mathcal{J}_+ = \emptyset$. For each group, the objective function drives all positive z_i to the value of

$$z_i = \bar{b}_i \quad i \in \mathcal{J}_+$$

while the remaining z_i , $i \in \mathcal{J}_-$ are pushed to the hypersphere constraint. Since the objective gradients are equal for all variables, all negative components have the same optimal value. Their value follows directly from the activity of the hypersphere constraint and is equal to

$$z_i = -\sqrt{\frac{1 - \sum_{j \in \mathcal{J}_+} \bar{b}_j^2}{m - n}} \quad i \in \mathcal{J}_-$$

The equality constraints then give $z_{i+m} = z_i$, $i = 1, \dots, m$ with which the optimal designs are complete. The objective function value for group n is equal to

$$f_n^* = \sum_{i \in \mathcal{J}_+} \bar{b}_i - (m - n) \sqrt{\frac{1 - \sum_{i \in \mathcal{J}_+} \bar{b}_i^2}{m - n}}$$

Table 2 Example 2, $m = 5$

	Algorithm					
	fmincon	LGO	ALC		ALC block	
	Local	Global	Local	Global	Local	Global
$f^* = -2.24$	32	10	0	10	0	10
$f^* = -1.73$	153	0	0	0	0	0
$f^* = -1.20$	316	0	1	0	1	0
$f^* = -0.63$	317	0	4	0	4	0
$f^* = 0.00$	151	0	0	0	0	0
$f^* = 1.12$	31	0	0	0	0	0
NC	0	0	95	0	95	0
Total	1000	10	100	10	100	10
Subproblem optimizations	–	–	34	37	19	15
Function evaluations	117	183,993	1–10: 195	1–10: 89,990	1–10: 92 0: 1,133	1–10: 33,247 0: 6,539

Number of starting points that converged to a specific objective function value for different algorithms, average number of required function evaluations and subproblem optimizations. The function evaluations are listed averaged over all 10 subproblems, where subproblem “0” is the convex master problem of the block-separable coordination variant. NC = did not converge to a feasible solution

When $n = 0$, all variables are negative such $\mathcal{J}_+ = \emptyset$, and we obtain the global optimum $f_0^* = -\sqrt{m}$. Since there are multiple combinations for which variables are positive or negative for group n , the total of local solutions in each group is $\binom{m}{n}$, and $\sum_{n=0}^m \binom{m}{n} = 2^m$.

A similar distribution of local solutions is observed from the results for $m = 5$ from 1000 different starting points, which are reported in Table 2. The global search with LGO converged to the global optimum for all ten runs. For $m = 10$ and $m = 20$, LGO experiences difficulties in identifying the global optimum, most likely caused by the many local minima for these cases (Table 3).

For the first set of distributed optimization experiments, Problem (17) is partitioned into $2m$ subsystems, one for each variable such that $\mathbf{x}_j = [z_j]$. Subsystems $j = 1, \dots, m$ have local objective $f_j = F_j = 2z_j$, and subsystems $j = m + 1, \dots, 2m$ have local objective $f_j = F_j = -z_j$. All subsystems have a local constraint $\mathbf{g}_j = [g_j]$, and constraints $\mathbf{g}_0 = [g_{2m+1}]$, and $\mathbf{h}_0 = [h_1, \dots, h_m]$ are the coupling constraints of the partition. The problem does not have any linking variables $\mathbf{y} = []$. Due to the non-convexity of the local constraints each subsystem has two disconnected feasible domains, and is therefore multi-modal.

The above partition resembles practical situations in which competing objectives between subsystems are present. Coordination methods have to determine which one is more important. Here, the objectives of subsystems j and $j + m$ are opposite: Subsystem j aims at minimizing z_j , while Subsystem $j + m$ wants to maximize z_{j+m} . The coupling constraint $h_j = z_j - z_{j+m} = 0$ however forces their values to be equal resulting in a competitive situation. The goal of coordination then

is to determine which objective is critical, and which should be sacrificed.

Two ALC variants are used to coordinate the partitioned problem. The first relaxes the coupling constraints $\mathbf{g}_0 = [g_{2m+1}]$ and $\mathbf{h}_0 = [h_1, \dots, h_m]$ in the traditional ALC fashion of Section 2.2 (labeled ALC). The slack variable s for the inequality constraint g_{2m+1} is included in the optimization variables of subsystem 1. The second variant (labeled ALC-block) follows an alternating direction approach of Section 2.3, and defines

Table 3 Example 2, $m = 10$ and $m = 20$

	All-in-one	ALC	ALC block
$m = 10$			
$f^* = -3.16$	2	10	10
$f^* = -2.80$	8	0	0
Total	10	10	10
Subproblem optimizations	–	45	16
Function evaluations	838,217	109,985	34,370 0: 13,484
$m = 20$			
$f^* = -4.47$	0	10	10
$f^* = -3.71$	6	0	0
$f^* = -3.45$	2	0	0
$f^* = -3.19$	2	0	0
Total	100	10	10
Subproblem optimizations	–	49	18
Function evaluations	1,681,222	119,694	38,257 0: 28,333

Number of starting points that converged to a specific objective function value for different algorithms, average number of required function evaluations and subproblem optimizations. The function evaluations are listed averaged over all $2m$ subproblems, and subproblem “0” is the convex master problem of the block-separable coordination variant ALC

responses $\mathbf{r}_j(\mathbf{x}_j) = \mathbf{x}_j$ for each subproblem. A master problem of the form (13) that solves for the support variables \mathbf{t}_j associated with the responses \mathbf{r}_j is introduced, and includes $\mathbf{g}_0 = [g_{2m+1}]$ and $\mathbf{h}_0 = [h_1, \dots, h_m]$ as local constraints.

Both coordination variants are illustrated in Fig. 4 by taking $k = 1$ (The value of k represents the number of variables per subproblems, and its use is explained later in this section). Again subproblems are solved either with a local search with `fmincon` or by performing a global search with LGO.

Results for $m = 5$ are listed in Table 2. They show that both ALC variants with only a local search at subproblems failed to converge to consistent solutions for most of the starting points. The solutions that did converge to a local solution started from an initial design where z_i and z_{i+m} had the same sign for each $i = 1, \dots, m$. The local constraint gradients point away from zero driving the subproblem solutions into the same feasible domain (either below $-a_i\sqrt{1/m}$ or above $b_i\sqrt{1/m}$). When one of the z_i and z_{i+m} pairs has opposite signs, the subproblems are pushed into different feasible regions (one below $-a_i\sqrt{1/m}$ and the other above $b_i\sqrt{1/m}$). The local search behavior of the subproblem solver keeps solutions within these different regions, and ALC cannot find a consistent solution. Solving the subproblems for global optimality removes this barrier, and all solutions converged to the global optimum.

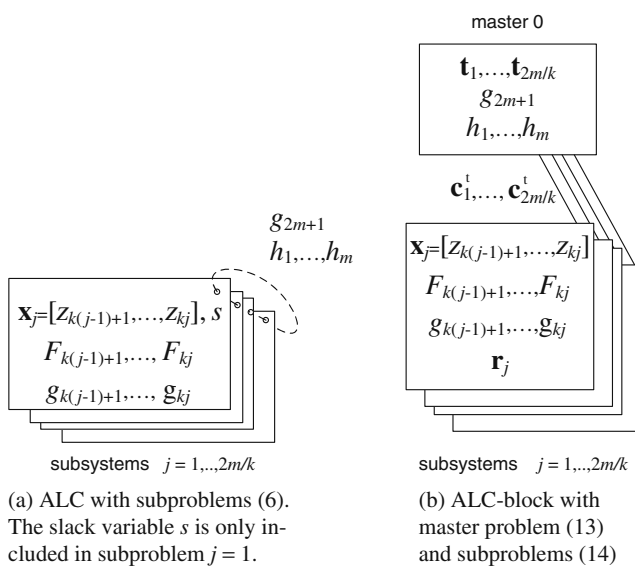


Fig. 4 Example 2: two ALC coordination variants. Dashed lines in the left figure indicate coupling through coupling constraints g_{2m+1} and h_1, \dots, h_m , and solid lines in the right figure indicate coupling through consistency constraints $\mathbf{c}_j^t = \mathbf{t}_j - \mathbf{r}_j(\mathbf{x}_j) = \mathbf{0}$. The problem has no linking variables $\mathbf{y} = []$ (a, b)

To compare computational cost for the global all-in-one approach with the global distributed approach, the all-in-one problem is also solved for $m = 10$ and $m = 20$ (Table 3). For all three values of m , the decomposition approach requires less function evaluations than the all-in-one approach using LGO. Moreover, the all-in-one runs do not converge to the global solution, while ALC consistently converges to the global optimum. The cost increase for going from $m = 5$ to $m = 10$ and from $m = 10$ to $m = 20$ are much smaller for ALC than the cost increase for LGO. For LGO, costs are roughly doubled for each step, while for ALC only 10% additional cost are required due to the relatively efficient coordination process.

A second set of experiments is set up to investigate the trade-off between subproblem size and computational cost for this example. To do so, the number of variables per subproblem is varied while keeping the total number of variables fixed by taking $m = 20$. Instead of a single variable, each subproblem solves for a block of k consecutive variables, where $k = 2, 5, 10$, and the total number of subproblems is $2m/k$. Each partition is solved with the two augmented Lagrangian coordination variants depicted in Fig. 4, and a global search is performed at each subproblem. Note that the results of Table 3 refer to a partition with $k = 1$.

Table 4 displays the obtained solutions and total number of function evaluations. For most experiments, the augmented Lagrangian coordination algorithms obtained the globally optimal solution. Only the runs for the traditional ALC variant for $k = 10$ converged to a number of local solutions. This behavior is most likely caused by the local search nature of the coordination algorithms together with the LGO subproblem solver, which was not able to identify the globally optimal solution for each subproblem.

Remarkably, the number of subproblem optimizations remained constant although the number of subproblems reduces for larger k . The number of linking constraints remains the same, which suggests that these are the main driver for coordination cost, and not the number of subproblems.

For larger values of k , the number of function evaluations increases. The main cause is the increased number of variables and constraints in each subproblem, which to a large extent influences the computational cost for the LGO algorithm (also observed for the all-in-one experiments). However, only the traditional ALC variant for $k = 10$ shows an increase in computational cost when compared to the all-in-one results; the remaining experiments all required less function evaluations than all-in-one. These findings suggest that decomposition is beneficial for this particular example

Table 4 Example 2, $m = 20$, and $k = 2, 5, 10$

Objective	$k = 2$		$k = 5$		$k = 10$	
	ALC	ALC block	ALC	ALC block	ALC	ALC block
$f^* = -4.47$	10	10	10	10	0	10
$f^* = -4.22$	0	0	0	0	4	0
$f^* = -3.97$	0	0	0	0	3	0
$f^* = -3.71$	0	0	0	0	3	0
Total	10	10	10	10	10	10
Subproblem optimizations	49	18	49	18	56	18
Subproblem evaluations	224,078	71,167	564,185	177,028	2,920,180	787,362
		0: 28,053		0: 27,831		0: 27,568

Number of starting points that converged to a specific objective function value for different algorithms, average number of required function evaluations and subproblem optimizations. The function evaluations are listed averaged over all $2m/k$ subproblems, and subproblem “0” is the convex master problem of the block-separable coordination variant

both in terms of solution quality and computational cost.

4.3 Example 3

With this third example, we demonstrate that the starting point and the sequence in which subproblems are solved may have an influence on which solution is obtained, even when subproblems are solved for global optimality. The problem has four variables, one equality constraint, and one inequality constraint. The objective and equality constraint are both non-convex such that the problem does not meet the assumptions of the convergence proof for ALC. The problem is given by

$$\begin{aligned}
 \min_{z_1, z_2, z_3, z_4} f &= \sum_{i=1}^3 F_i = 4 \left(z_1 - \frac{3}{2} \right) \left(z_3 - \frac{1}{2} \right) \\
 &\quad - (z_2 - 1)^2 - 2(z_4 - 1)^2 \\
 \text{s.t. } g &= -2z_1 - z_2 + \frac{3}{4} \leq 0 \\
 h &= -z_3 z_4 + 1 = 0 \\
 0 &\leq z_i \leq 2, \quad i = 1, \dots, 4
 \end{aligned} \tag{18}$$

where $F_1 = 4(z_1 - \frac{3}{2})(z_3 - \frac{1}{2})$, $F_2 = -(z_2 - 1)^2$ and $F_3 = -2(z_4 - 1)^2$.

All 10 runs of the all-in-one experiments using the global search variant of LGO converged to an optimal value of $f^* = -10\frac{1}{2}$, with $\mathbf{z}^* = [0, 2, 2, \frac{1}{2}]$ and all constraints active. The average number of function evaluations reported by LGO was 8,270. The 1000 all-in-one runs with fmincon converged to three local minima: the global solution, a local solution $\mathbf{z}^* = [0, \frac{3}{4}, 2, \frac{1}{2}]$ with $f^* = -9\frac{9}{16}$, and a solution with $f^* = -3$ that is unique in variables $z_3^* = \frac{1}{2}$ and $z_4^* = 2$. Variable z_2 is either 0 or 2, and z_1 can take any value as long as the inequality constraint g is satisfied. The average number of function evaluations for the converged runs was 203.

Two partitions are selected for Problem (18). The first partition, ALC1, has two subsystems that are coupled only through the objective term $f_0 = F_1$. Subsystem 1 of this partition has local variables $\mathbf{x}_1 = [z_1, z_2]$, a local objective $f_1 = F_2$, and a local inequality constraint $\mathbf{g}_1 = [g]$. Subsystem 2 has local variables $\mathbf{x}_2 = [z_3, z_4]$, a local objective $f_2 = F_3$, and a local equality constraint $\mathbf{h}_2 = [h]$. The second partition, ALC2, is similar to the first but treats variable z_3 as a linking variable $\mathbf{y} = [z_3]$. As a result, the objective F_1 can be included in subsystem 1: $f_0 = []$ and $f_1 = F_1 + F_2$. Both partitions are illustrated in Fig. 5.

The first partition is coordinated using the traditional ALC variant of Section 2.2, and the second partition is coordinated using the central master problem of Section 2.3, similar to the second variant for the previous example.

Table 5 displays the results for the distributed optimization experiments. These results again show the ability of augmented Lagrangian coordination to find solutions to the all-in-one problem. Similar to the first example, the solutions obtained with a local search at

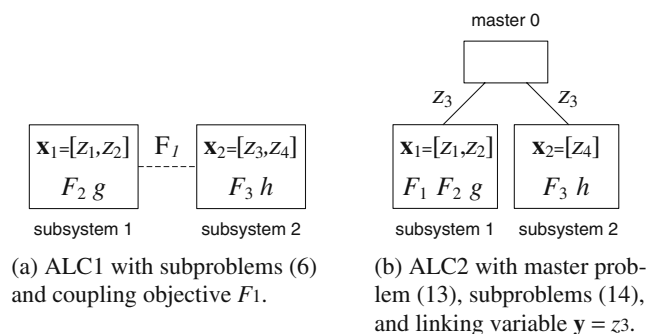


Fig. 5 Example 3: two ALC coordination variants. *Dashed lines* in the left figure indicate coupling through coupling objective $f_0 = F_1$, and *solid lines* indicate coupling through consistency constraints of linking variable $\mathbf{y} = z_3$ (a, b)

Table 5 Example 3

	Algorithm					
	fmincon	LGO	ALC1		ALC2	
	Local	Global	Local	Global	Local	Global
Objective $f^* = -10\frac{1}{2}$	334	10	31	8	20	10
$f^* = -9\frac{9}{16}$	332	0	31	0	24	0
$f^* = -3$	334	0	38	2	44	0
Other ($-3.94 \leq f^* \leq -2.46$)	0	0	0	0	12	0
Total	1000	10	100	10	100	10
Subproblem optimizations	–	–	3	3	22	8
Function evaluations	26	8,270	1: 30	1: 6,538	1: 386	1: 24,661
			2: 26	2: 10,290	2: 159	2: 28,548
					0: 166	0: 95

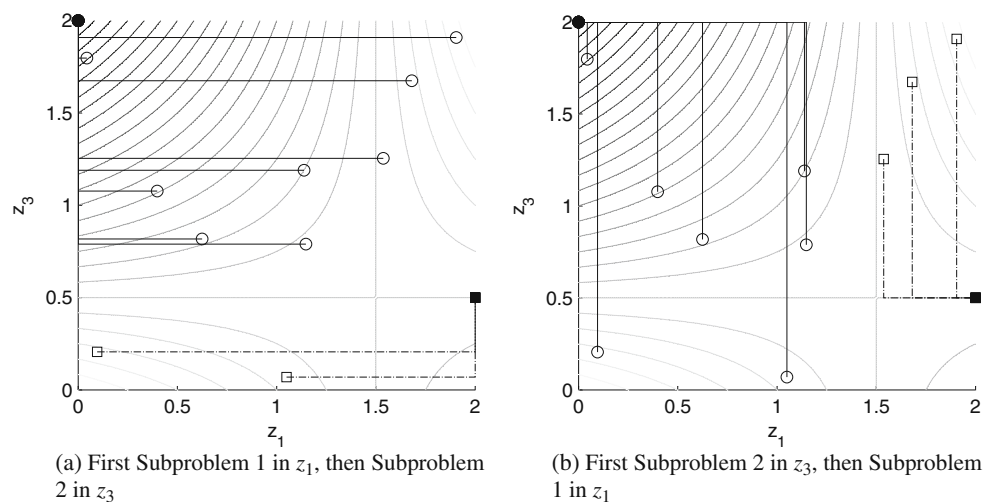
Number of starting points that converged to a specific objective function value for different algorithms, average number of required function evaluations and subproblem optimizations. The function evaluations are listed per subproblem, where subproblem “0” is the convex master problem of the block-separable coordination variant

subproblems are similar to those obtained with the all-in-one implementation using a local search. Some of the starting points are scattered wider around the local solution with $f^* = -3$ then one would expect from the termination tolerance of $\varepsilon = 10^{-2}$. This scattering is most likely caused by numerical inaccuracies due to subproblem solver tolerances.

In contrast to the previous example, performing a global search at each subproblem for ALC1 does not yield the global solution for each starting point. For two out of the ten runs only a local solution was obtained, indicating that the starting point does have an effect on which solution is obtained when using a global search. The influence of the starting point is caused by the non-convexity of the coupling objective $f_0 = F_1 = 4(z_1 - \frac{3}{2})(z_3 - \frac{1}{2})$. Contour plots of this function are depicted in Fig. 6, together with the iteration history for each starting point. The figure clearly shows that the value of initial value of z_3 (of the

second subproblem) determines to which intermediate solution $[z_1, z_2]$ Subproblem 1 converges. For $z_3 > \frac{1}{2}$ (initial designs marked with a circle \circ), the derivative of f_0 with respect to z_1 becomes positive such that Subproblem 1 aims at minimizing $az_1 + 3 - (z_2 - 1)^2$ subject to $g = -2z_1 - z_2 + \frac{3}{4} \leq 0$, where $a = 4z_3 - 2 > 0$. The globally optimal solution to Subproblem 1 then becomes $[z_1^*, z_2^*] = [0, 2]$. For these values, Subproblem 2 aims at minimizing $-6z_3 + 3 - 2(z_4 - 1)^2$ subject to $h = -z_3z_4 + 1 = 0$ which yields $[z_3^*, z_4^*] = [2, \frac{1}{2}]$. Since $z_3 > \frac{1}{2}$, the solution to Subproblem 1 does not change, and we have converged to the solution $\mathbf{z}^* = [0, 2, 2, \frac{1}{2}]$ with $f^* = -10\frac{1}{2}$. Following a similar procedure shows that if the initial value for $z_3 < \frac{1}{2}$ (square markers \square) converges to $\mathbf{z}^* = [2, 0, \frac{1}{2}, 2]$ with $f^* = -3$. For $z_3 = \frac{1}{2}$, the solution to Subproblem 1 is non-unique in z_1 and the value for z_1^* that is returned by the solver determines to which solution ALC converges. Note that the local minimum with $f^* = -9\frac{9}{16}$ with an initial

Fig. 6 Example 3: Contour plots of the coupling objective $f_0 = F_1 = 4(z_1 - \frac{3}{2})(z_3 - \frac{1}{2})$ and iteration paths for ALC1 with global search. Both the starting point and the order of the sequence in which subproblems are solved determine which solution is obtained (**a, b**)



value for $z_3 > \frac{1}{2}$ is eliminated by the global search at Subproblem 1.

An additional factor that determines to which solution ALC1 converges is the sequence in which Subproblems 1 and 2 are solved. If one solves Subproblem 2 first, then the initial value for z_1 determines to which solution ALC converges. Figure 6 shows the iteration history for each starting point when subproblems are solved in this reverse order. The figure shows that an initial design that has $z_1 < \frac{3}{2}$ converges to $\mathbf{z}^* = [0, 2, 2, \frac{1}{2}]$ with $f^* = -10\frac{1}{2}$, and $z_1 > \frac{3}{2}$ yields $\mathbf{z}^* = [2, 0, \frac{1}{2}, 2]$ with $f^* = -3$.

An important observation is that the solutions obtained with Partition ALC2 do not depend on the initial design or solution sequence for this example. For Partition ALC2, the non-convex objective f_0 is local to Subproblem 1, and subproblems are only coupled through the convex augmented Lagrangian penalty term ϕ on the inconsistency with respect to the linking variable z_3 . For this particular partition, performing a global search at subproblems suffices to obtain the global solution.

The computational cost for Partition ALC1 with a global search at the subproblems are similar to those for the all-in-one implementation, indicating that the computational cost for ALC can be comparable with respect to those for an all-in-one implementation for this example. The computational cost for Partition ALC2 are higher than the all-in-one implementation and Partition ALC1. The differences between the two partitions are mainly caused by the fact that Partition ALC1 has a coupling objective, while Partition ALC2 has a coupling variable. For the coupling objective, no external penalty update loop is required, and the ALC algorithm therefore terminates rapidly. The coupling variable for the second partition does require an outer loop to set the penalty parameters, thereby incurring additional computational cost. For the local search experiments, the all-in-one implementation, with a termination tolerance of 10^{-6} , already outperforms the distributed optimization approaches that have a much looser termination tolerance of 10^{-2} .

4.4 Example 4

The fourth example demonstrates that for some sequences of the inner loop, a global search at subproblems yields only a locally optimal solution for every starting point, even though the coupling functions are linear. The example is a version of the third test problem of Deb (2000) with different upper bounds and variable z_{13} eliminated from the problem since its maximization was only constrained by its upper bound.

The resulting problem has twelve variables, a concave objective function, and nine linear inequality constraints. The convergence proofs for the ALC algorithms are valid for this example since the constraint sets are convex.

$$\begin{aligned} \min_{\mathbf{z}} f(\mathbf{z}) &= \sum_{i=1}^{12} F_i(z_i) \\ &= \sum_{i=1}^4 (5z_i - 5z_i^2) + \sum_{i=5}^{12} (-z_i) - 3 \\ \text{subject to } g_1 &= 2z_1 + 2z_2 + z_{10} + z_{11} - 10 \leq 0 \\ g_2 &= 2z_1 + 2z_3 + z_{10} + z_{12} - 10 \leq 0 \\ g_3 &= 2z_2 + 2z_3 + z_{11} + z_{12} - 10 \leq 0 \\ g_4 &= -8z_1 + z_{10} \leq 0 \\ g_5 &= -8z_2 + z_{11} \leq 0 \\ g_6 &= -8z_3 + z_{12} \leq 0 \\ g_7 &= -2z_4 - z_5 + z_{10} \leq 0 \\ g_8 &= -2z_6 - z_7 + z_{11} \leq 0 \\ g_9 &= -2z_8 - z_9 + z_{12} \leq 0 \\ 0 &\leq z_i \leq 3 \quad i = 1, \dots, 12 \end{aligned} \quad (19)$$

All 10 runs of the all-in-one experiments using the global search variant of LGO converged to an optimal value of $f^* = -104\frac{1}{4}$, with $\mathbf{z}^* = [2\frac{1}{2}, 2\frac{1}{2}, 2\frac{1}{2}, 3, 3, 3, 3, 3, 0, 0, 0]$ with the first three constraints active. The average number of function evaluations reported by LGO was 97,881. The 1000 all-in-one runs with `fmincon` converged to a large number of local minima, including the one obtained with the global search. Table 6 reports the optimal values of local minima that were obtained from at least ten of the starting points. The solutions obtained less than ten times have optimal values between -59.95 and -23.48 , and are reported as “other” in Table 6. The average number of function evaluations for the converged runs was 68.

Problem (19) is partitioned into three subsystems. The first subsystem has local variables $\mathbf{x}_1 = [z_1, z_4, z_5, z_{10}]$, local objective $f_1 = F_1 + F_4 + F_5 + F_{10} - 1$, and local constraints $\mathbf{g}_1 = [g_4, g_7]$. The second subsystem has local variables $\mathbf{x}_2 = [z_2, z_6, z_7, z_{11}]$, local objective $f_2 = F_2 + F_6 + F_7 + F_{11} - 1$, and local constraints $\mathbf{g}_2 = [g_5, g_8]$. The third subsystem has local variables $\mathbf{x}_3 = [z_3, z_8, z_9, z_{12}]$, local objective $f_3 = F_3 + F_8 + F_9 + F_{12} - 1$, and local constraints $\mathbf{g}_3 = [g_6, g_9]$. Constraints $\mathbf{g}_0 = [g_1, g_2, g_3]$ depend on variables of all

Table 6 Example 4

	Algorithm						
	fmincon	+ AL	LGO	ALC		ALC-block	
	Local	Local	Global	Local	Global	Local	Global
$f^* = -104.25$	425	79	10	86	0	53	10
$f^* = -89.83$	281	0	0	0	10	11	0
$f^* = -89.75$	16	0	0	0	0	0	0
$f^* = -87.52$	0	0	0	0	0	16	0
$f^* = -81.91$	72	4	0	0	0	6	0
$f^* = -81.70$	30	0	0	0	0	0	0
$f^* = -74.25$	88	17	0	14	0	6	0
$f^* = -59.83$	61	0	0	0	0	0	0
$f^* = -51.91$	10	0	0	0	0	2	0
Other $-59.95 \leq f^* \leq -23.48$	17	0	0	0	0	6	0
Total	1000	100	10	100	10	100	10
Subproblem optimizations	–	–	–	40	32	30	25
Function evaluations	68	986	97,881	1: 1,122	1: 396,505	1: 398	1: 118,553
				2: 635	2: 156,690	2: 396	2: 121,065
				3: 616	3: 149,785	3: 300	3: 121,088
						0: 399	0: 489

Number of starting points that converged to a specific objective function value for different algorithms, average number of required function evaluations and subproblem optimizations. The function evaluations are listed per subproblem, where subproblem “0” is the convex master problem of the block-separable coordination variant

three subsystems and are coupling constraints of the problems. The problem has no linking variables $\mathbf{y} = []$.

The above partition is coordinated using two ALC variants. The first variant, labeled ALC, coordinates the three subsystems in a traditional ALC fashion following Section 2.2. The slack variables $\mathbf{s} = [s_1, s_2, s_3]$ for the coupling constraints $\mathbf{g}_0 = [g_1, g_2, g_3]$ are assigned to Subsystem 1 such that $\mathbf{s}_1 = [s_1, s_2, s_3]$, and $\mathbf{s}_2 = \mathbf{s}_3 = []$.

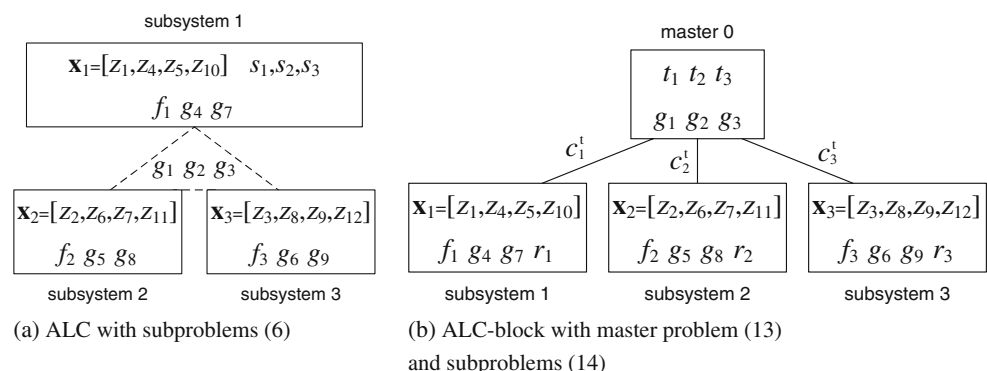
The second variant uses the fact that the coupling constraints $\mathbf{g}_0 = [g_1, g_2, g_3]$ are block-separable with $r_j(\mathbf{x}_j) = [2z_j + z_{j+9} - 5]$, $j = 1, 2, 3$, such that $g_1 = r_1 + r_2$, $g_2 = r_1 + r_3$, and $g_3 = r_2 + r_3$. Since the coupling constraints are block-separable, an alternating direction approach of Section 2.3 is selected to coordinate the decomposed problem for this variant. A master

problem (13) is included that has support variables $\mathbf{t}_j = [t_j]$, $j = 1, 2, 3$ as its design variables and constraints $\mathbf{g}_0 = [g_1, g_2, g_3]$ as constraints. The master problem is linked to the subproblems through a penalty on the consistency constraints $c_j^t = t_j - r_j(\mathbf{x}_j) = 0$, $j = 1, 2, 3$. Both coordination variants are depicted in Fig. 7.

Table 6 displays the results for the distributed optimization experiments. The experiments for the first ALC variant show that the coordination algorithms have a preference for certain local solutions for this example. ALC with a local search at the subproblems converged to only two local solutions, one of which is the global solution. This particular preference is caused by the augmented Lagrangian relaxation of the coupling constraints $\mathbf{g}_0 = [g_1, g_2, g_3]$. If these constraints are also

Fig. 7 Example 4: two ALC coordination variants.

Dashed lines in the left indicate coupling through coupling constraints $\mathbf{g}_0 = [g_1, g_2, g_3]$, and solid lines in the right figure indicate coupling through consistency constraints $c_j^t = t_j - r_j(\mathbf{x}_j) = 0$



handled through an augmented Lagrangian approach in the all-in-one implementation, a very similar preference for the two local solutions obtained with ALC is observed (Table 6, column “+ AL”). Apparently, the differences between the algorithmic treatment of the coupling constraints between ALC (augmented Lagrangian relaxation) and *fmincon* (linearization) causes the differences between the distribution of obtained solutions.

The experiments for traditional ALC with a global search at subproblems show a preference for a local solution with $f^* = -89.83$, which however is not the global optimum. For this case, the treatment of constraints does not explain why only a local optimum is obtained, since an all-in-one augmented Lagrangian-based implementation with a global search consistently converges to the global solution $f^* = -104.25$. Similar to the second example, the sequence in which the subproblems are solved in the inner loop determines which solution is obtained. If the inner loop sequence is changed to 3, 2, 1, instead of 1, 2, 3, then the global solution is obtained for each of the ten starting points. How to determine which sequence yields the best solution remains an open question.

The preference for certain local solutions is less pronounced for the ALC-block experiments with a local search. An explanation for this may be the fact that it is not the original constraints of the problem that are relaxed, but the consistency constraints. ALC-block with a global search again converges to the global optimum for each of the ten runs.

The computational cost for both ALC variants with a global search at subproblems are in general about 50% higher than for the all-in-one approach. The larger number of function evaluations for Subproblem 1 in the first partition, ALC, are again caused by the three additional slack variables associated with the coupling constraints $\mathbf{g}_0 = [g_1, g_2, g_3]$. The results for using only a local search at subproblems show a factor 5–15 increase in computational cost when compared to the all-in-one implementation with *fmincon*. Although a single subproblem optimization requires fewer function evaluations than the solution of the all-in-one problem, the number of iterations required in the coordination process increases the total cost for the distributed approach above those observed for the non-decomposed implementation.

4.5 Example 5

The purpose of this fifth example is to demonstrate that ALC algorithms may fail for problems with dis-

connected feasible domains, even when a global search is employed at each subproblem. The problem has two variables, a concave objective, and one equality constraint, while each variable is restricted to be within three disconnected regions:

$$\begin{aligned} \min_{z_1, z_2} \quad & f = -(z_1 - 4)^2 - (z_2 - 4)^2 \\ \text{s.t.} \quad & h = z_1 - z_2 = 0 \\ & z_1 \in [1, 2] \cup [3, 4] \cup [8, 9] \\ & z_2 \in [\tfrac{1}{2}, 2] \cup [2\tfrac{1}{2}, 3\tfrac{1}{2}] \cup [9\tfrac{1}{2}, 10] \end{aligned} \quad (20)$$

The globally optimal solution is $\mathbf{z}^* = [1, 1]$ with $f^* = -18$. The problem has a second local minimum at $\mathbf{z}^* = [3, 3]$ with $f^* = -2$.

Although the above example is not particularly challenging as an all-in-one implementation, its disconnected feasible domain proves to be a challenge for ALC algorithms. To illustrate these difficulties, the problem is partitioned into two subsystems, one associated with each variable: Subsystem 1 with $\mathbf{x}_1 = [z_1]$ and $f_1 = -(z_1 - 4)^2$, and Subsystem 2 with $\mathbf{x}_2 = [z_2]$ and $f_1 = -(z_2 - 4)^2$. Then constraint h becomes a linear coupling constraint $\mathbf{h}_0 = [h]$. If the partition is solved using a traditional ALC algorithm with a global search at subproblems, then ALC iterates will not converge to a feasible solution from *any* starting point. Instead, ALC converges to the infeasible point $\mathbf{z} = [9, 9\tfrac{1}{2}]$ that violates the coupling constraint.

This behavior is caused by the quadratic part of the augmented Lagrangian penalty function $w^2(z_1 - z_2)^2$ in combination with the alternating minimization inner loop. For large penalty weights, the quadratic part puts a large penalty on any point that is far away from the solution to the other subproblem, thereby introducing a local search behavior. This local search causes a preference for designs that are optimal with the respect to the all-in-one problem *without* the coupling constraint h , since violations of this constraint are initially very small. The global solution to the all-in-one problem without constraint h is $\mathbf{z} = [9, 10]$, and this point is obtained after the first iteration. This point however violates the coupling constraint h , and the penalty parameters v and w are both updated in the outer loop. The updates of v and w drive the values of z_1 and z_2 closer together to reduce the constraint violations, until the unfeasible point $\mathbf{z} = [9, 9\tfrac{1}{2}]$ is reached. In the direct neighborhood of this design however, no point with a smaller constraint violation can be found. Ideally, the Lagrange multiplier v should push iterates towards feasible designs, but the quadratic penalty part prohibits this. Although the Lagrange multiplier grows larger, so does the penalty

weight w , which keeps the iterates at the infeasible point $\mathbf{z} = [9, 9\frac{1}{2}]$, and ALC gets trapped.

It is not enough to simply place an upper bound \bar{w} on the value of the penalty weight w to resolve this difficulty. Although the linear part will eventually push iterates into a feasible region, the ALC procedure terminates prematurely at a non-optimal design. At these designs, the large quadratic part avoids a more detailed search within the feasible region. This premature convergence due to large penalty weights has been observed in earlier work regarding ALC and analytical target cascading (Tosserams et al. 2006, 2007). Note that the effect of premature convergence was not observed for Examples 2 and 6, which also have disconnected feasible domains.

4.6 Example 6

The final example is the portal frame design optimization problem originally introduced by Sobieszczanski-Sobieski et al. (1985), and used by many other researchers in the context of coordination methods (see, e.g., Vanderplaats et al. 1990; Arslan and Hajela 1997; Liu et al. 2004). It is included here as an illustration of the use of ALC in structural optimization.

The problem aims at finding the cross-sectional dimensions of a three-beam portal frame, illustrated in Fig. 8a, that is subjected to a horizontal force and a concentrated moment. Although the portal frame problem has been used extensively, no single study reports all implementation details necessary for a reproduction of the presented results. For example, variable bounds are often not reported although they have a large effect on the optimal solution. For the sake of completeness and reproducibility, our implementation details are included in this section, and necessary analysis equations are presented in the Appendix.

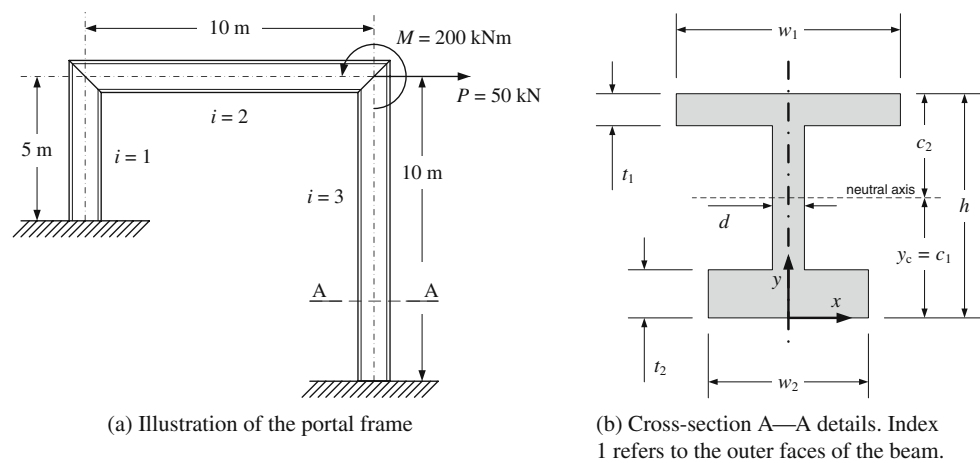
We follow the general formulation of the problem given by Liu et al. (2004). The design problem aims at finding the 18 cross sectional variables ($\mathbf{z} = [\mathbf{z}^{[1]}, \mathbf{z}^{[2]}, \mathbf{z}^{[3]}]$), 6 for each beam ($\mathbf{z}^{[i]} = [h^{[i]}, w_1^{[i]}, w_2^{[i]}, d^{[i]}, t_1^{[i]}, t_2^{[i]}]$, see Fig. 8b for definitions), such that the volume of the structure is minimized. Here, the top-right index refers to one of the three beams. Limits are posed on normal and shear stresses in all beams (g_1 – g_4). Geometry constraints on the beam cross sections (g_5 – g_7) are used to prevent slender structures that are likely to buckle. Following Liu et al. (2004), constraints g_8 require that either the top flange area is twice as large as the bottom one, or vice versa. These constraints effectively separate the feasible domain into

eight disconnected regions thereby introducing at least as many local minima.

The original portal frame design problem is given by:

$$\begin{aligned} & \min_{\mathbf{z}} V(\mathbf{z}) \\ & \text{subject to} \\ & g_{1,i,j,k}(\mathbf{z}) = \frac{\sigma_{j,k}^{[i]}}{\sigma_y} - 1 \leq 0 \\ & \quad i = 1, 2, 3 \quad j = 1, 2 \quad k = 1, 2 \\ & g_{2,i,j,k}(\mathbf{z}) = -\frac{\sigma_{j,k}^{[i]}}{\sigma_y} - 1 \leq 0 \\ & \quad i = 1, 2, 3 \quad j = 1, 2 \quad k = 1, 2 \\ & g_{3,i,j}(\mathbf{z}) = \frac{\tau_j^{[i]}}{\tau_y} - 1 \leq 0 \\ & \quad i = 1, 2, 3 \quad j = 1, 2 \\ & g_{4,i,j}(\mathbf{z}) = -\frac{\tau_j^{[i]}}{\tau_y} - 1 \leq 0 \\ & \quad i = 1, 2, 3 \quad j = 1, 2 \\ & g_{5,i}(\mathbf{z}^{[i]}) = \frac{h^{[i]} - t_1^{[i]} - t_2^{[i]}}{35d^{[i]}} - 1 \leq 0 \\ & \quad i = 1, 2, 3 \\ & g_{6,i,j}(\mathbf{z}^{[i]}) = \frac{w_j^{[i]}}{20t_j^{[i]}} - 1 \leq 0 \\ & \quad i = 1, 2, 3 \quad j = 1, 2 \\ & g_{7,i}(\mathbf{z}^{[i]}) = 1 - \frac{5(h^{[i]} - t_1^{[i]} - t_2^{[i]})d^{[i]}}{A^{[i]}} \leq 0 \\ & \quad i = 1, 2, 3 \\ & g_{8,i}(\mathbf{z}^{[i]}) = \left(1 - \frac{t_1^{[i]}w_1^{[i]}}{2t_2^{[i]}w_2^{[i]}}\right) \left(1 - \frac{t_2^{[i]}w_2^{[i]}}{2t_1^{[i]}w_1^{[i]}}\right) \leq 0 \\ & \quad i = 1, 2, 3 \\ & \mathbf{z}_{\min} \leq \mathbf{z} \leq \mathbf{z}_{\max} \end{aligned} \quad (21)$$

where the lower and upper bounds on the variables are $15 \text{ cm} \leq w_1, w_2 \leq 150 \text{ cm}$, $20 \text{ cm} \leq h \leq 200 \text{ cm}$, $1.0 \text{ cm} \leq d \leq 10 \text{ cm}$, and $0.75 \text{ cm} \leq t_1, t_2 \leq 7.5 \text{ cm}$. Constraints g_1 and g_2 limit the normal stress σ in the three beams ($i = 1, 2, 3$) for both beam ends ($j = 1, 2$) at the top and bottom free surfaces ($k = 1, 2$). Since the normal stress attains its maximum in one of these locations, constraints g_1 and g_2 effectively assure that the normal stresses are below the allowed value $\sigma_y = 200 \cdot 10^6 \text{ N/m}^2$ throughout the whole structure. Similarly, constraints g_3 and g_4 limit the shear stress τ at

Fig. 8 Portal frame design problem (a, b)

the neutral axis in the three beams ($i = 1, 2, 3$) at both ends ($j = 1, 2$), which assures that the shear stress does not exceed the allowed value $\tau_y = 116 \cdot 10^6$ N/m² at any other location. Constraint g_5 limits the aspect ratio of the web of each beam to be smaller than 35, while constraint g_6 limits the aspect ratio of the flanges to be below 20. Constraint g_7 assures that the area of the web is at least 20% of the total cross section area A . Constraint g_8 requires that either the top flange area is twice as large as the bottom flange area, or vice versa. The equations required to determine the stresses are given in the [Appendix](#).

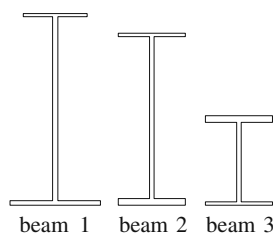
All 10 runs of the all-in-one experiments using the global search variant of LGO converged to an optimal value (rounded) of $f^* = 0.1661$ with beams 1 and 2 with larger bottom flanges, and beam 3 with a larger top flange. Details for this design can be found in [Fig. 9](#). The average number of function evaluations for the ten initial designs reported by LGO was 246,738. The 1000 all-in-one runs with `fmincon` converged to a large number of local minima, including the one obtained

with the global search.³ The average number of function evaluations for the converged runs was 628. [Table 7](#) reports the optimal values of the obtained local minima.

In general, solutions can be divided into two groups. Group 1 typically has the cross-sectional dimensions of beams 1 and 2 at its lower bounds and of beam 3 larger, while Group 2 has beams 1 and 2 larger and beam 3 small. A number of local minima are observed within each group, where each minimum depends on which flange is chosen larger for which beam. A qualitative comparison with existing results for the portal frame example shows that the observed trends in the local minima can also be found in the results reported in literature. The results of Arslan and Hajela (1997) and Schutte et al. (2004) have beams 1 and 2 small and beam 3 larger, similar to the first group. The results given by Sobieszczanski-Sobieski et al. (1985), Vanderplaats et al. (1990), and Liu et al. (2004) have beam 3 small and beams 1 and 2 larger, similar to the second group.

The problem is partitioned in a traditional multi-level fashion (see, e.g. Sobieszczanski-Sobieski et al. 1985; Haftka and Gürdal 1993) by defining three subsystems associated with detailed design of each beam, and a single system-level subsystem concerned with the overall design of the portal frame in terms of the cross sectional areas and moments of inertia $\mathbf{x}_1 = [A^{[1]}, A^{[2]}, A^{[3]}, I^{[1]}, I^{[2]}, I^{[3]}]$. The top subsystem has no local objective and no local constraints. The variable bounds are computed from the bounds for the original variables \mathbf{z} , and are set to $0.41 \cdot 10^{-2} \text{ m}^2 \leq A \leq 0.41 \text{ m}^2$, and $0.26 \cdot 10^{-4} \text{ m}^4 \leq I \leq 0.26 \text{ m}^4$.

Global solution: $V^* = 0.1661 \text{ m}^3$			
	beam 1	beam 2	beam 3
A	88.0	71.1	51.5
h	45.0	38.4	20.0
w_1	15.0	15.0	15.0
w_2	21.2	15.0	15.0
d	1.23	1.03	1.00
t_1	0.75	0.75	1.50
t_2	1.06	1.50	0.75

**Fig. 9** Global solution characteristics (beam dimensions in cm and areas in cm²)

³Note that a termination tolerance of 10^{-8} was necessary to obtain a clear separation between objective values. Looser tolerances cause the solutions to “overflow” in each other.

Table 7 Example 6

	Algorithm			
	fmincon	LGO	ALC	
	Local	Global	Local	Local-global
Group 1				
$f^* = 0.1661$	55	10	42	10
$f^* = 0.1671$	63	0	0	0
$f^* = 0.1684$	50	0	12	0
$f^* = 0.1687$	51	0	0	0
$f^* = 0.1694$	46	0	0	0
$f^* = 0.1695$	50	0	0	0
$f^* = 0.1699$	53	0	0	0
$f^* = 0.1710$	53	0	0	0
$f^* = 0.1790$	16	0	0	0
Group 2				
$f^* = 0.1684$	236	0	45	0
$f^* = 0.1688$	210	0	0	0
Other				
$0.1666 < f^* < 0.1755$	7	0	1	0
NC	110	0	0	0
Total	1000	10	100	10
Subproblem optimizations	–	–	620	342
Function evaluations	1,442	246,738	1: 54,401 2: 269,329 3: 268,275 4: 227,442	1: 4,887,939 2: 862,761 3: 514,665 4: 456,202

Number of starting points that converged to a specific objective function value for different algorithms, average number of required function evaluations and subproblem optimizations. The function evaluations are listed per subproblem

At the beam subsystems $j = 2, 3, 4$, support variables for the six member forces $\mathbf{f}^{[i]}$, $i = 1, 2, 3$ are introduced to determine the stresses in the beams. Each of the three beam subsystems designs one beam for its detailed cross-sectional dimensions $\mathbf{z}^{[i]}$. Together with the additionally introduced variables for the member forces $\mathbf{f}^{[i]}$, $i = 1, 2, 3$, this gives for the local variables of subsystem: $\mathbf{x}_{i+1} = [\mathbf{z}^{[i]}, \mathbf{f}^{[i]}]$, $i = 1, 2, 3$. Local constraints are the stress and geometry constraints $\mathbf{g}_{i+1} = [g_{1,i}(\mathbf{x}^{[i]}), \dots, g_{8,i}(\mathbf{x}^{[i]})]$. The subsystems have their individual volume as local objectives $f_{i+1} = L_i A(\mathbf{z}^{[i]})$. For $\mathbf{z}^{[i]}$, the original bounds are used, and the bounds for the member forces are $-10^6 \text{ N(m)} \leq \mathbf{f} \leq 10^6 \text{ N(m)}$, where forces are in Newton and moments in Newton-meters.

Eighteen equality constraints $\mathbf{h}_f = [\mathbf{h}_f^{[1]}, \mathbf{h}_f^{[2]}, \mathbf{h}_f^{[3]}] = \mathbf{0}$ are introduced to couple the member force variables $\mathbf{f}^{[i]}$ of each beam to the values $\mathbf{f}_{\text{fem}}^{[i]}(\mathbf{x}_1)$ computed at the system level, where $\mathbf{h}_f^{[i]} = \mathbf{f}^{[i]} - \mathbf{f}_{\text{fem}}^{[i]}(\mathbf{x}_1)$. Six equality constraints $\mathbf{h}_a = [\mathbf{h}_a^{[1]}, \mathbf{h}_a^{[2]}, \mathbf{h}_a^{[3]}] = \mathbf{0}$ are introduced to link the areas and the moments of inertia $A^{[i]}, I^{[i]}$ used at the system-level problem to the values associated with the detailed cross-section dimensions $A_{\text{beam}}^{[i]}(\mathbf{x}_{i+1})$, $I_{\text{beam}}^{[i]}(\mathbf{x}_{i+1})$, where $\mathbf{h}_a^{[i]} = [A^{[i]} - A_{\text{beam}}^{[i]}(\mathbf{x}_{i+1}), I^{[i]} - I_{\text{beam}}^{[i]}(\mathbf{x}_{i+1})]$, where the area $A_{\text{beam}}^{[i]}(\mathbf{x}_{i+1})$ and moment of inertia $I_{\text{beam}}^{[i]}(\mathbf{x}_{i+1})$ of beam i are given by (22)–(23). These introduced constraints couple the beam subsystems to the system level, and are therefore the coupling equality constraints

of the problem: $\mathbf{h}_0(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \mathbf{x}_4) = [\mathbf{h}_0^{[1]}, \mathbf{h}_0^{[2]}, \mathbf{h}_0^{[3]}]$ with $\mathbf{h}_0^{[i]}(\mathbf{x}_1, \mathbf{x}_{i+1}) = [\mathbf{h}_f^{[i]}, \mathbf{h}_a^{[i]}]$. The partition structure is depicted in Fig. 10 and has no linking variables $\mathbf{y} = []$. The partitioned problem is coordinated with a distributed ALC formulation of Section 2.2 following the partition structure.

The above partition requires the introduction of 24 (!) support variables. Since the number of variables is a driving factor in the cost of coordination for ALC, elimination of these variables by substitution of the

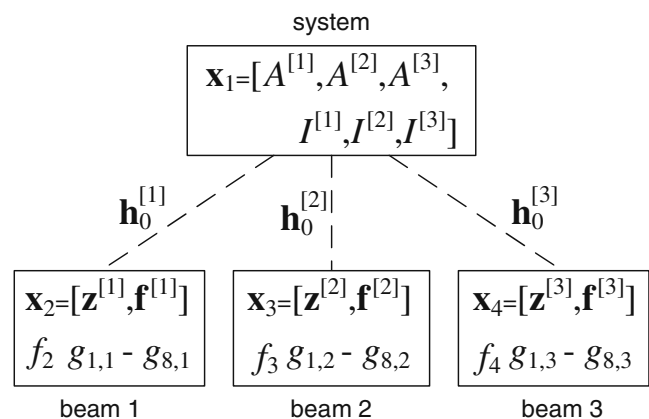


Fig. 10 Portal frame: partition structure. Dashed lines indicate coupling through the coupling constraints $\mathbf{h}_0 = [\mathbf{h}_0^{[1]}, \mathbf{h}_0^{[2]}, \mathbf{h}_0^{[3]}]$. The problem has no linking variables $\mathbf{y} = []$

coupling constraints is attractive. Doing so would however render all design constraints to become coupling constraints. These constraints therefore have to be relaxed explicitly, and appear in multiple subproblems. For example, if the member force variables are eliminated, the stress constraints then depend on responses of the system-level analysis equations, and have to be included at both the system level and the lower level. To maintain a larger degree of autonomy, the partition with the 24 support variables is used. The numerical experiments demonstrate that this has consequences for the computational cost.

The default settings for augmented Lagrangian coordination do not produce a clear enough distinction between the local minima. To obtain a clear distinction, we use a stricter termination tolerance of $\varepsilon = 10^{-4}$, and a more moderate value for $\beta = 1.01$.

Furthermore, we do not perform a global search using LGO at the beam subproblems. If we were to use LGO, over 100 subproblem optimizations that take around half an hour on a desktop PC each are necessary. To avoid such excessive cost, the beam subproblems are not solved using a full global search. Instead, a search with LGO's local variant is performed *for each flange choice*. The design with the lowest objective is then selected as the optimal design for that beam to achieve some sense of global optimality at the lower level. The system-level problem only has six variables, and is therefore solved with a global search using LGO. A second set of experiments, labeled ALC-local, is performed that performs a local search with `fmincon` at the system-level subproblem and for each flange choice of the beam subproblems.

The results reported in Table 7 show that ALC-local converges to local solutions of the all-in-one problem. ALC reached the best all-in-one optimum from each starting point when a global search using LGO is performed at the system level (ALC-local-global). When only a local search is performed (ALC-local), the obtained ALC solutions are optimal within their group for the majority of the starting points. The local search for each flange choice appears to be an efficient way to introduce a global search-like behavior at the beam subproblems.

Solution costs for ALC are much higher than those for the all-in-one implementations. The 24 additional support variables, the tight termination tolerance, and the small value for β all contribute to these high cost. Convergence can be sped up by increasing the value of β , however this causes the algorithm to “stall” at non-optimal designs due to the large penalty weights. These large penalty weights slow down convergence of the alternating minimization loop, and causes the

iterates to lie close together (ultimately within the tolerance margin) even when still far away from an optimal solution.

The large differences in function evaluations at the beam subproblems between the two ALC variants are caused by the different local solvers that are used. In the first ALC variant, Matlab's SQP implementation is used, while the second ALC variant uses the generalized reduced gradient implementation of LGO. The difference in efficiency of these solvers explains the differences in computational cost between the two ALC variants.

5 Summary of observations

Augmented Lagrangian coordination (ALC) algorithms are provably convergent to local solutions of the non-decomposed problems under smoothness and convexity assumptions. For the non-convex and multi-modal problems studied in this article, ALC was often able to converge to appropriate solutions, even through the requirements for convergence are not met. The solutions obtained by performing a local search at ALC subproblems are similar to those obtained from an all-in-one implementation with a local search for the majority of the examples. As can be expected, ALC algorithms with a local search did not perform well for problems with disconnected feasible domains.

Performing a global search at subproblems yielded globally optimal solutions to the non-decomposed problem for many examples, especially for problems with block-dependent coupling functions. Although this observation is encouraging, Example 5 demonstrated that performing a global search at subproblems may fail to produce even a feasible solution for some problems. Furthermore, the numerical results clearly show the well-known trend in global optimization that performing a global search yields better solutions, but at much higher computational cost. The question whether these cost are justified by the improved solutions is just as valid for ALC as it is for the all-in-one case.

The numerical experiments revealed three major characteristics of ALC algorithms that have an influence on its convergence behavior. First, the alternating minimization inner loop is the largest contributor to the convergence behavior of ALC. For the example with a multi-modal coupling objective, both the starting point as well as the sequence in which subproblems are solved determines which local solution is obtained, even when subproblems are solved for global optimality. The second factor of influence is the augmented Lagrangian relaxation of the linking constraints. By initially relaxing

these constraints, ALC algorithms that perform only a local search at subproblems can show a preference for certain solutions. When a global search is performed at subproblems, this relaxation does not play a role for the examples presented here. Where the first two factors are inherited from the existing algorithms used in the inner and outer loops, the third factor comes from the specific combination of these techniques. For problems with disconnected feasible domains, ALC algorithms may converge to solutions that are infeasible, but that are close to the optimal solution with all coupling constraints relaxed. For these problems, the quadratic part of the augmented Lagrangian penalty acts as a tight trust region, causing the ALC algorithms to terminate prematurely at a point that does not satisfy one or more coupling constraints.

For global searches, cost savings through problem decomposition using ALC appear possible for partitions with subproblems that are a few factors smaller in dimension than the original problem. Although such partitions may require a relatively large coordination effort, the costs for solving the subproblems are much lower than for the original problem, resulting in an overall decrease in the total number of function evaluations. For local searches, these cost benefits are not observed, even when subproblems are much lower in dimension than the original problem.

6 Concluding remarks

The examples presented in this article show that especially the alternating minimization inner loop is a limiting factor for achieving global optimality with an augmented Lagrangian coordination approach. We expect that the analytical target cascading method (ATC, Kim 2001) may experience similar difficulties since it is a subclass of ALC. In fact, the difficulties associated with disconnected feasible domains have been observed for ATC by Kim and Papalambros (2006). Other coordination approaches that use alternating minimization such as MDOIS (Shin and Park 2005), BLISS (Sobieszcanski-Sobieski et al. 2000), and enhanced collaborative optimization (Roth and Kroo 2008) may also be affected by the behavior of the sequential process.

It is worthwhile to investigate whether coordination methods that follow a nested structure instead of an alternating one may be less influenced by non-convexities and multi-modality of the subproblems. Examples of nested approaches are collaborative optimization (Braun et al. 1997), BLISS2000 (Sobieszcanski-Sobieski et al. 2003), quasi-separable

decomposition (Haftka and Watson 2005), and the penalty decomposition methods of DeMiguel and Murray (2006). However, performing a global search at the system level in a nested structure is unattractive since for each function evaluation at this level, a full global search is required at the subproblem level. An option to reduce the high cost for such a nested approach is to decouple the system-level optimization from the lower-level optimizations through the use of response surfaces (see, e.g., Liu et al. 2004). Alternatively, one can look at coordination methods that define separable subproblems such as the Lagrangian decomposition method proposed by Blouin et al. (2005).

Finally, examples and numerical results presented in this paper can serve as a benchmark for comparison of other coordination methods on multi-modal problems. The problems are challenging in terms of the multi-modality, but simple enough to allow an analysis of the inner workings of coordination algorithms. Moreover, Example 2 may also be an interesting, scalable test problem for the global optimization community.

Appendix: Analysis equations for the portal frame structural optimization problem

Cross-section properties

The area $A^{[i]}$ and moment of inertia $I^{[i]}$ of beam i are given by (indices i dropped for brevity of notation):

$$A = w_1 t_1 + (h - t_1 - t_2)d + w_2 t_2 \quad (22)$$

$$\begin{aligned} I = & \frac{w_1(t_1)^3}{12} + w_1 t_1 \left(y_c - \left(h - \frac{1}{2} t_1 \right) \right)^2 \\ & + \frac{d(h - t_1 - t_2)^3}{12} + d(h - t_1 - t_2) \left(y_c - \frac{1}{2} h \right)^2 \\ & + \frac{w_2(t_2)^3}{12} + w_2 t_2 \left(y_c - \frac{1}{2} t_2 \right)^2 \end{aligned} \quad (23)$$

where the centroid y_c is given by:

$$y_c = \frac{w_1 t_1 (h - \frac{1}{2} t_1) + \frac{1}{2} (h - t_1 - t_2) d h + \frac{1}{2} w_2 (t_2)^2}{w_1 t_1 + d(h - t_1 - t_2) + w_2 t_2} \quad (24)$$

The first moment of area Q of the material above the neutral axis for a beam, relevant in computing the shear stress, is given by:

$$Q = w_1 t_1 \left(h - y_c - \frac{1}{2} t_1 \right) + \frac{1}{2} d (h - y_c - t_1)^2 \quad (25)$$

The free surface distances c_1 and c_2 , required for determining the maximal bending stress, are given by:

$$c_1 = h - y_c \quad c_2 = y_c \quad (26)$$

at the faces of flanges 1 (outer) and 2 (inner), respectively.

Assembled stiffness matrix

To determine the stresses in the frame, first the forces and moments acting on the individual beams have to be computed. To this end, we use a finite element analysis that consists of three plane bending elements, each associated with one I beam of the frame. Each end $j = 1, 2$ of a beam element is subjected to an axial force N_j , a shear force V_j , and a bending moment M_j . The positive directions in each beam are defined as illustrated in Fig. 11.

The vector of member forces \mathbf{f} is given by: $\mathbf{f} = [N_1, V_1, M_1, N_2, V_2, M_2]^T$, and the vector of local member node displacements \mathbf{u} is given by $\mathbf{u} = [u_1, v_1, \theta_1, u_2, v_2, \theta_2]^T$, where u_j , v_j , and θ_j are the axial and perpendicular displacements, and the rotation of beam end j , respectively. The positive directions of the displacements coincide with the positive directions of the member forces.

The assembled stiffness matrix \mathbf{K}_{ass} of the frame is constructed from the local stiffness matrices of each beam. The local stiffness matrix that relates the mem-

ber forces to the member displacements (in the local coordinate system) through $\mathbf{f} = \mathbf{K}_{\text{local}}\mathbf{u}$ is given by:

$$\mathbf{K}_{\text{local}} = \begin{bmatrix} \frac{EA}{L} & 0 & 0 & -\frac{EA}{L} & 0 & 0 \\ 0 & \frac{12EI}{L^3} & \frac{6EI}{L^2} & 0 & -\frac{12EI}{L^3} & \frac{6EI}{L^2} \\ 0 & \frac{6EI}{L^2} & \frac{4EI}{L} & 0 & -\frac{6EI}{L^2} & \frac{2EI}{L} \\ -\frac{EA}{L} & 0 & 0 & \frac{EA}{L} & 0 & 0 \\ 0 & -\frac{12EI}{L^3} & -\frac{6EI}{L^2} & 0 & \frac{12EI}{L^3} & -\frac{6EI}{L^2} \\ 0 & \frac{6EI}{L^2} & \frac{2EI}{L} & 0 & -\frac{6EI}{L^2} & \frac{4EI}{L} \end{bmatrix} \quad (27)$$

To create the assembled stiffness matrix, first the element stiffness matrices have to be rotated from the local to the global coordinate system. To this end, a rotation matrix $\mathbf{R} = \mathbf{R}(\alpha)$ is used such that $\mathbf{K} = (\mathbf{R})^T \mathbf{K}_{\text{local}} \mathbf{R}$, where the matrix $\mathbf{R} = \mathbf{R}(\alpha)$ is given by:

$$\mathbf{R} = \mathbf{R}(\alpha) = \begin{bmatrix} \cos(\alpha) & \sin(\alpha) & 0 & 0 & 0 & 0 \\ -\sin(\alpha) & \cos(\alpha) & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & \cos(\alpha) & \sin(\alpha) & 0 \\ 0 & 0 & 0 & -\sin(\alpha) & \cos(\alpha) & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (28)$$

The rotation angles α for the three beams are: $\alpha^{[1]} = \pi/2$, $\alpha^{[2]} = 0$, and $\alpha^{[3]} = -\pi/2$.

To determine the assembled stiffness matrix \mathbf{K}_{ass} for the whole structure, the local nodes are mapped to the global nodes, and the rotated local stiffness matrices are inserted at the appropriate places of the assembled stiffness matrix \mathbf{K}_{ass} . To this end, a projection matrix $\mathbf{P}^{[i]}$ for beam i can be defined such that:

$$\mathbf{K}_{\text{ass}} = \sum_{i=1}^3 (\mathbf{P}^{[i]})^T \mathbf{K}^{[i]} \mathbf{P}^{[i]} \quad (29)$$

For beams 1, 2, and 3 the projection matrices are given by:

$$\mathbf{P}^{[1]} = \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} \\ \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix} \quad \mathbf{P}^{[2]} = \begin{bmatrix} \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} \end{bmatrix} \quad (30)$$

$$\mathbf{P}^{[3]} = \begin{bmatrix} \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} \end{bmatrix}$$

with \mathbf{I} the 3×3 identity matrix and $\mathbf{0}$ a 3×3 zero matrix.

Unknown displacements and reaction forces

To compute the unknown displacements and reaction forces required to compute member stresses, we consider the boundary conditions of the construction. From the boundary conditions we know that

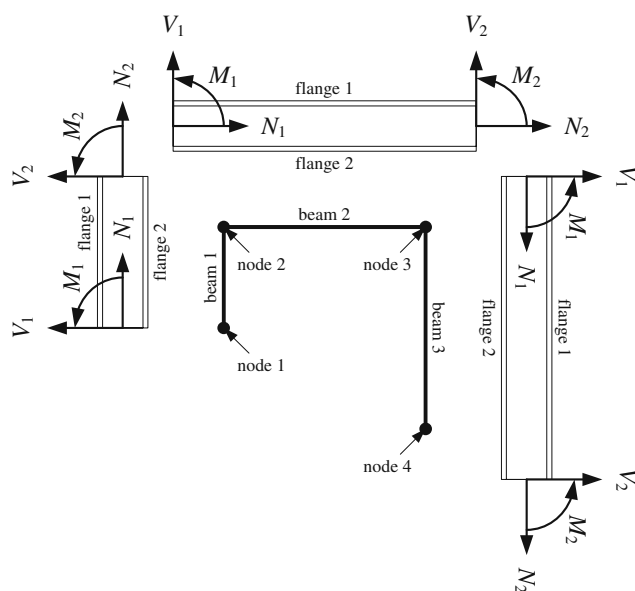


Fig. 11 Free body diagrams of portal frame beams

nodes 1 and 4 are fixed $[u_1, v_1, \theta_1]^T = [u_4, v_4, \theta_4]^T = [0, 0, 0]^T$, and the reaction forces at nodes 2 and 3 are known: $[X_2, Y_2, Z_2]^T = [0, 0, 0]^T$ and $[X_3, Y_3, Z_3]^T = [50 \cdot 10^3, 0, 200 \cdot 10^3]^T$ (see Fig. 11 for the definition of the node numbers). The remaining displacements $[u_2, v_2, \theta_2]^T$ and $[u_3, v_3, \theta_3]^T$, and reaction forces $[X_1, Y_1, Z_1]^T$ and $[X_4, Y_4, Z_4]^T$ are unknown.

To compute these unknowns, the displacement and force vectors are split up into two parts: a known part (boundary conditions) and an unknown part (free nodes), such that $\mathbf{u}_{\text{ass}} = [\mathbf{u}_{\text{free}}^T, \mathbf{u}_{\text{bc}}^T]^T$ and $\mathbf{f}_{\text{ass}} = [\mathbf{f}_{\text{bc}}^T, \mathbf{f}_{\text{free}}^T]^T$, where $\mathbf{u}_{\text{free}} = [u_2, v_2, \theta_2, u_3, v_3, \theta_3]^T$, $\mathbf{u}_{\text{bc}} = [u_1, v_1, \theta_1, u_4, v_4, \theta_4]^T = [0, 0, 0, 0, 0, 0]^T$, $\mathbf{f}_{\text{bc}} = [X_2, Y_2, Z_2, X_3, Y_3, Z_3]^T = [0, 0, 0, 50 \cdot 10^3, 0, 200 \cdot 10^3]^T$, and $\mathbf{f}_{\text{free}} = [X_1, Y_1, Z_1, X_4, Y_4, Z_4]^T$.

Under these conventions, the unknowns can be computed by the system of equations $\mathbf{K}_{\text{ass}} \mathbf{u}_{\text{ass}} = \mathbf{f}_{\text{ass}}$, which is given by:

$$\begin{bmatrix} \mathbf{K}_{11} & \mathbf{K}_{12} \\ \mathbf{K}_{12}^T & \mathbf{K}_{22} \end{bmatrix} \begin{bmatrix} \mathbf{u}_{\text{free}} \\ \mathbf{u}_{\text{bc}} \end{bmatrix} = \begin{bmatrix} \mathbf{f}_{\text{bc}} \\ \mathbf{f}_{\text{free}} \end{bmatrix} \quad (31)$$

where \mathbf{K}_{11} , \mathbf{K}_{12} , and \mathbf{K}_{22} are 6×6 submatrices of the assembled stiffness matrix \mathbf{K}_{ass} .

From this system, and because $\mathbf{u}_{\text{bc}} = 0$, the free displacements are given by:

$$\mathbf{u}_{\text{free}} = \mathbf{K}_{11}^{-1}(\mathbf{f}_{\text{bc}} - \mathbf{K}_{12}\mathbf{u}_{\text{bc}}) = \mathbf{K}_{11}^{-1}\mathbf{f}_{\text{bc}} \quad (32)$$

The unknown reaction forces $\mathbf{f}_{\text{free}} = [X_1, Y_1, Z_1, X_4, Y_4, Z_4]^T$ are given by:

$$\mathbf{f}_{\text{free}} = \mathbf{K}_{12}^T \mathbf{u}_{\text{free}} + \mathbf{K}_{22} \mathbf{u}_{\text{bc}} = \mathbf{K}_{12}^T \mathbf{K}_{11}^{-1} \mathbf{f}_{\text{bc}} \quad (33)$$

Computation of stresses

The nodal forces $\mathbf{f}^{[i]}$ in beam i , required for computing the stresses, are given by (indices i dropped again):

$$\mathbf{f} = [N_1, V_1, M_1, N_2, V_2, M_2]^T = \mathbf{K}_{\text{local}} \mathbf{u} = \mathbf{K}_{\text{local}} \mathbf{R} \mathbf{P} \mathbf{u}_{\text{ass}} \quad (34)$$

The axial stress σ_a is constant throughout a beam and is given by:

$$\sigma_a = \frac{N_2 - N_1}{A} \quad (35)$$

The bending stresses $\sigma_{b,j,k}$ at end $j = 1, 2$ in the top ($k = 1$) and bottom ($k = 2$) flanges of beam i , respectively, are given by:

$$\begin{aligned} \sigma_{b,1,1} &= \frac{c_1 M_1}{I} & \sigma_{b,1,2} &= -\frac{c_2 M_1}{I} \\ \sigma_{b,2,1} &= -\frac{c_1 M_2}{I} & \sigma_{b,2,2} &= \frac{c_2 M_2}{I} \end{aligned} \quad (36)$$

The four normal stresses $\sigma_{j,k}$ at flanges $k = 1, 2$ of beam ends $j = 1, 2$ are given by:

$$\sigma_{j,k} = \sigma_a + \sigma_{b,j,k} \quad j = 1, 2 \quad k = 1, 2 \quad (37)$$

The shear stresses τ_j at the neutral axis at the beams ends $j = 1, 2$ are given by:

$$\tau_j = \frac{V_j Q}{I d} \quad j = 1, 2 \quad (38)$$

Open Access This article is distributed under the terms of the Creative Commons Attribution Noncommercial License which permits any noncommercial use, distribution, and reproduction in any medium, provided the original author(s) and source are credited.

References

- Alexandrov NM, Lewis RM (2002) Analytical and computational aspects of collaborative optimization for multidisciplinary design. *AIAA J* 40(2):301–309
- Allison JT, Papalambros PY (2008) Consistency constraint allocation in augmented lagrangian coordination. In: *Proceedings of the ASME design engineering technical conferences*, Brooklyn, New York, 3–6 August 2008
- Arora JS, Chahande AI, Paeng JK (1991) Multiplier methods for engineering optimization. *Int J Numer Methods Eng* 32(7):1485–1525
- Arsalan MA, Hajela P (1997) Counterpropagation neural networks in decomposition based optimal design. *Comput Struct* 65(5):641–650
- Balling RJ, Sobieszczanski-Sobieski J (1996) Optimization of coupled systems: a critical overview of approaches. *AIAA J* 34(1):6–17
- Bertsekas DP (1982) *Constrained optimization and lagrange multiplier methods*. Academic, New York
- Bertsekas DP (2003) *Nonlinear programming*, 2nd edn, 2nd printing. Athena Scientific, Belmont
- Bertsekas DP, Tsitsiklis JN (1989) *Parallel and distributed computation*. Prentice-Hall, Englewood Cliffs
- Bezdek JC, Hathaway R (2002) Some notes on alternating optimization. *Lect Notes Comput Sci* 2275:288–300
- Blouin VY, Lassiter JB, Wiecek MM, Fadel GM (2005) Augmented Lagrangian coordination for decomposed design problems. In: *Proceedings of the 6th world congress on structural and multidisciplinary optimization*, Rio de Janeiro, 30 May–3 June 2005
- Braun RD (1996) *Collaborative optimization: an architecture for large-scale distributed design*. PhD thesis, Stanford University
- Braun RD, Moore AA, Kroo IM (1997) Collaborative approach to launch vehicle design. *J Spacecr Rockets* 34(4):478–486
- Cramer EJ, Dennis JE, Frank PD, Lewis RM, Shubin GR (1994) Problem formulation for multidisciplinary optimization. *SIAM J Optim* 4(4):754–776
- Deb K (2000) An efficient constraint handling method for genetic algorithms. *Comput Methods Appl Mech Eng* 186(2–4):311–338
- DeMiguel AV, Murray W (2000) An analysis of collaborative optimization methods. In: *Proceedings of the 8th*

- AIAA/USAF/NASA/ISSMO symposium on multidisciplinary analysis and optimization, AIAA paper 2000-4720, Long Beach, 6–8 September 2000
- DeMiguel AV, Murray W (2006) A local convergence analysis of bilevel decomposition algorithms. *Optim Eng* 7(2):99–133
- Fortin M, Glowinski R (1983) Augmented Lagrangian methods: application to the numerical solution of boundary-value problems. North-Holland, Amsterdam
- Gill PE, Murray W, Wright MH (1981) Practical optimization. Academic, London
- Grippo L, Sciandrone M (2000) On the convergence of the block nonlinear Gauss-Seidel method under convex constraints. *Oper Res Lett* 26(3):127–136
- Haftka RT, Gürdal Z (1993) Elements of structural optimization, 3rd edn. Kluwer Academic, Dordrecht
- Haftka RT, Watson LT (2005) Multidisciplinary design optimization with quasiseparable subsystems. *Optim Eng* 6(1):9–20
- Kim HM (2001) Target cascading in optimal system design. PhD thesis, University of Michigan
- Kim HM, Papalambros PY (2006) Target exploration for disconnected feasible regions in enterprise-driven multilevel product design. *AIAA J* 44(1):67–77
- Kim HM, Michelena NF, Papalambros PY, Jiang T (2003) Target cascading in optimal system design. *ASME J Mech Des* 125(3):474–480
- LGO Solver Suite (2008) Pinter consulting. <http://www.pinterconsulting.com>. Accessed 30 June 2008
- Li Y, Lu Z, Michalek JJ (2008) Diagonal quadratic approximation for parallelization of analytical target cascading. *ASME J Mech Des* 131(5):1–11
- Lin J (2004) Analysis and enhancement of collaborative optimization for multidisciplinary design. *AIAA J* 42(2):348–360
- Liu B, Haftka RT, Watson LT (2004) Global-local structural optimization using response surfaces of local optimization margins. *Struct Multidiscipl Optim* 27(5):352–359
- Mathworks (2008) Matlab user guide, Version 7. Mathworks, Natick
- Michelena N, Kim HM, Papalambros PY (1999) A system partitioning and optimization approach to target cascading. In: Proceedings of the 12th international conference on engineering design, Munich, 24–26 August 1999
- Michelena NF, Park H, Papalambros PY (2003) Convergence properties of analytical target cascading. *AIAA J* 41(5):897–905
- Roth BD, Kroo IM (2008) Enhanced collaborative optimization: a decomposition-based method for multidisciplinary design. In: Proceedings of the ASME design engineering technical conferences, Brooklyn, New York, 3–6 August 2008
- Schutte JF, Haftka RT, Watson LT (2004) Decomposition and two-level optimization of structures with discrete sizing variables. In: Proceedings of the 45th AIAA/ASME/ASCE-AHS/ASC structures, structural dynamics, and materials conference, AIAA paper 2004-1541, Palm Springs, 19–22 April 2004
- Shin MK, Park GJ (2005) Multidisciplinary design optimization based on independent subspaces. *Int J Numer Methods Eng* 64(5):599–617
- Sobieski IP, Kroo I (2000) Collaborative optimization using response surface estimation. *AIAA J* 38(10):1931–1938
- Sobieszczanski-Sobieski J (1988) Optimization by decomposition: A step from hierarchic to non-hierarchic systems. In: Proceedings of the 2nd NASA air force symposium on advances in multidisciplinary analysis and optimization, NASA CP-3031, Hampton, September 1988
- Sobieszczanski-Sobieski J, James BB, Dovi AR (1985) Structural and multidisciplinary optimization by multilevel decomposition. *AIAA J* 23(11):1775–1782
- Sobieszczanski-Sobieski J, Agte JS, Sandusky Jr RR (2000) Bilevel integrated system synthesis. *AIAA J* 38(1):164–172
- Sobieszczanski-Sobieski J, Altus TD, Phillips M, Sandusky Jr RR (2003) Bilevel integrated system synthesis for concurrent and distributed processing. *AIAA J* 41(10):1996–2003
- Tosserams S, Etman LFP, Papalambros PY, Rooda JE (2006) An augmented Lagrangian relaxation for analytical target cascading using the alternating direction method of multipliers. *Struct Multidiscipl Optim* 31(3):176–189
- Tosserams S, Etman LFP, Rooda JE (2007) An augmented Lagrangian decomposition method for quasi-separable problems in MDO. *Struct Multidiscipl Optim* 34(3):211–227
- Tosserams S, Etman LFP, Rooda JE (2008) Augmented Lagrangian coordination for distributed optimal design in MDO. *Int J Numer Methods Eng* 73(13):1885–1910
- Tosserams S, Etman LFP, Rooda JE (2009a) Block-separable linking constraints in augmented Lagrangian coordination. *Struct Multidiscipl Optim* 37(5):521–527
- Tosserams S, Etman LFP, Rooda JE (2009b) A classification of methods for distributed system optimization based on formulation structure. *Struct Multidiscipl Optim* doi:10.1007/s00158-008-0347-z
- Vanderplaats GN, Yang YJ, Kim DS (1990) Sequential linearization method for multilevel optimization. *AIAA J* 28(2):290–295